Scalable and Explainable Visually-Aware Recommender Systems

Thanet Markchom[a], Huizhi Liang[b,*], James Ferryman[a]

[a]*Department of Computer Science, University of Reading, Reading, RG6 6AH, UK*
[b]*School of Computing, Newcastle University, Newcastle upon Tyne, NE1 7RU, UK*

Abstract

Recommender systems are popularly used to deal with an information overload issue. Existing systems mainly focus on user-item interactions and semantic information derived from metadata of users and items to improve recommendation accuracy. Item images provide useful information to infer users' individual preferences, especially for those domains where visual factors are influential such as fashion items. However, this type of information has been ignored by most previous work. To bridge this gap and meet the requirements of performance from the aspects of Accuracy, Scalability, and Explainability evaluation metrics, this paper proposes a scalable and explainable visually-aware recommender system framework called SEV-RS. This framework contains a visually-augmented heterogeneous information network, a scalable meta-path feature extraction method for multi-hop relations, and a shallow explainable meta-path based Collaborative Filtering recommendation approach. We compared SEV-RS with the state-of-the-art models such as the deep learning model using Graph Attention Network on two real-world datasets and one synthetic dataset. The results show that SEV-RS produced more accurate and more explainable recommendations. Also, SEV-RS has substantially less computational time than the compared deep learning models.

Keywords: Recommender System, Heterogeneous Information Network, Meta-Path, Visual Information, Scalability, Explainability
2010 MSC: 00-01, 99-00

*Corresponding author

Notation	Description	Notation	Description
$\mathcal{U}$	users set	$\mathcal{P}_u$	set of items interacted by user u
$\mathcal{P}$	items set	$\mathcal{U}_p$	set of users interacted with item p
$\mathbb{G}$	HIN schema	$\mathbb{G}'$	visually-augmented HIN schema
$\mathbb{N}$	node types set	$\mathbb{N}'$	node types set in $\mathbb{G}'$
$\mathbb{R}$	relation types set	$\mathbb{R}'$	relation types set in $\mathbb{G}'$
$\mathbb{W}$	weight function of relation types in $\mathbb{R}$	$\mathbb{W}'$	weight function for relation types in $\mathbb{R}'$
$\mathcal{G}$	HIN	$\mathcal{G}'$	visually-augmented HIN
$\mathcal{N}$	nodes set	$\mathcal{N}'$	nodes set in $\mathcal{G}'$
$\mathcal{R}$	relations set	$\mathcal{R}'$	relations set in $\mathcal{G}'$
N_i/N_j	the ith/jth node type	$\mathcal{V}$	visual factor nodes set
$R_{N_iN_j}$	relation type from N_i to N_j	$\mathcal{R}_V$	visual relations set
$R^{-1}_{N_i,N_j}$	inverse relation type from N_j to N_i	V	visual factor node type
x, y	nodes	R_{UV}	user-visual factor relation type
$r_{x,y}$	relation from node x to y	R_{PV}	item-visual factor relation type
ϕ	node type mapping function	k_v	the number of visual factors
ψ	relation type mapping function	$\mathbf{v}_i$	the ith visual factor
$w(x, y)$	weight of relation $r_{x,y}$	v_i	the ith visual factor node of $\mathbf{v}_i$
m	meta-path	$\mathbf{v}^u$	the user visual preference profile
m'	probabilistic meta-path	$\hat{x}_{up}$	recommendation score of user u towards item p
δ	probability of probabilistic meta-path	α	global offset
z	path instance	β_u	user bias term
n_i	node with the type N_i	β_p	item bias term
$\mathcal{Z}_m$	set of path instances of m	$\boldsymbol{\gamma_u}$	user u traditional latent factors ($K_1 \times 1$)
$Pr(z)$	probability of path instance z	$\boldsymbol{\gamma_p}$	item p traditional latent factors ($K_1 \times 1$)
$s(u, p, m)$	meta-path based connectivity strength	$\boldsymbol{\theta_u}$	user u meta-path based latent factors ($K_2 \times 1$)
$a_{u,m}$	User-MetaPath association	$\boldsymbol{\theta_p}$	item p meta-path based latent factors ($K_2 \times 1$)
$a_{p,m}$	Item-MetaPath association	$\boldsymbol{\beta}_P$	item feature bias ($\lvert\mathcal{M}\rvert \times 1$)
$g(m)$	global connectivity of m	$\boldsymbol{\beta}_U$	user feature bias ($\lvert\mathcal{M}\rvert \times 1$)
$C(k, k+1)$	probability of N_{k+1} given N_k type nodes	$\mathbf{E}_U$	latent space projection matrix of f_u ($K_2 \times \lvert\mathcal{M}\rvert$)
$\mathcal{M}$	set of meta-paths	$\mathbf{E}_P$	latent space projection matrix of f_p ($K_2 \times \lvert\mathcal{M}\rvert$)
m_i	the ith meta-path in $\mathcal{M}$	$\mathbf{u}$	final user u latent factors ($(K_1 + K_2) \times 1$)
f_u	user meta-path feature	$\mathbf{p}$	final item p latent factors ($(K_1 + K_2) \times 1$)
f_p	item meta-path feature	σ	sigmoid function
E_{up}	explainability score between u and p	Θ	BPR-MF model parameters
$h(f_u, f_p)$	cosine similarity between f_u and f_p	λ_Θ	regularization hyper-parameter
$\mathcal{P}^+_u$	set of positive items of user u	λ_E	explainability regularization hyper-parameter
$\mathcal{D}_S$	training sample set		

Table 1: Notations

1. Introduction

Recently, there has been a large exponential growth in the information available for being consumed or selected by users. This can be problematic when enormous information is presented to them. To mitigate this issue, recommender systems suggest pieces of information (or items) that most potentially match each user's individual interests [1]. They have been popularly adopted on many online communities and platforms in various domains including e-commerce, e-health, and e-learning [2]. Existing recommender systems mainly focus on using user-item interactions to learn users' preferences. Since they only rely on user-item interactions, their performances drop drastically when such interactions are sparse or unavailable. To overcome these data sparsity and cold start problems [3], side information such as metadata of users and items has been used to enrich the connectivity between users and items, and improve recommendation accuracy. Besides semantic information derived from metadata, item images provide useful information to infer users' individual preferences. Since an image can provide numerous information compared to a single word, it is intuitively capable of providing rich information about users' preferences. For example, in clothing or fashion recommender systems, some users may prefer buying items with similar/complementary visual appearances rather than those that have the same category as their purchased items. Typically, an image contains several features which can be used for capturing users' visual preferences, for instance, shapes, textures, and colors. However, this type of information has been ignored by most existing work. How to better utilize item images and effectively profile users' individual visual preferences still need to be explored.

The performances of recommender systems have been mainly evaluated from the aspect of accuracy [4]. In spite of that, many current situations require other aspects to be jointly considered. In the era of big data, the ever-growing amount of information challenges the efficiency of recommendation generation. Scalability has become one of the important performance evaluation requirements when developing and applying recommender systems in the real world. Also, explainability has become an emerging performance evaluation metric, as required by the regulations such as the General Data Protection Regulation (GDPR) of the European Union and other countries. However, since this area is relatively new, the concept of explainability in recommender systems remains an open research question.

Both scalability and explainability have been individually considered in developing visually-aware recommender systems. Some attempts have been made to tackle a large-scale visually-aware recommendation problem [5, 6]. In terms

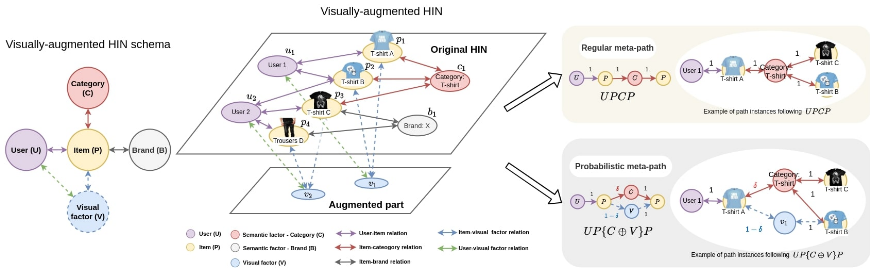


Figure 1: Example of visually-augmented HIN schema, visually-augmented HIN and the comparison of using the regular meta-path $UPCP$ and the probabilistic meta-path $UP\{C \oplus V\}P$

of explainability, most existing work was proposed to enable image-based explainable recommendations [7, 8, 9]. However, visually-aware recommender systems that consider both scalability and explainability have still been rather overlooked. To bridge the gap of profiling users' individual visual preferences effectively and meet the performance requirements of Accuracy, Scalability, and Explainability, this paper proposes a **S**calable and **E**xplainable **V**isually-aware **R**ecommender **S**ystem (SEV-RS) framework. Inspired by the omnipresence of heterogeneous information networks (HINs) [10] in explainable recommender systems, this work uses such networks to facilitate the task of visually-aware recommendation with explainability. However, many existing HIN-based recommendation frameworks typically ignored visual information and also suffered from the scalability issues [11, 12]. Thus, unlike existing approaches, our approach integrates visual elements into a HIN, extracts information from this HIN in a scalable way, and eventually uses this information to produce explainable visually-aware recommendations.

This proposed framework consists of three components. The first component is a **visually-augmented heterogeneous information network**. Typically, HINs are constructed based on only semantic factors derived from metadata of users and items without considering visual factors [13, 14, 15, 16, 3]. To incorporate visual factors, we introduce visual factor nodes generated from image features of items images and connect them to user and item nodes via visual relations. To utilize high-order relations in a HIN, meta-paths [14, 13] have been frequently adopted. However, as the importance of visual factors is usually different from semantic factors, we introduce *probabilistic meta-paths* to weight the importance of each factor.

The second component is a **scalable meta-path feature extraction** method. Based on visually-augmented HIN and probabilistic meta-paths, we extract meta-path features to profile users and items. Meta-path based approaches are popularly used to make HIN-based recommendations due to their capability of extracting semantically meaningful multi-hop relations [3]. However, it is challenging to develop effective and efficient methods for such approaches [13, 16, 3, 12]. For a length l meta-path, let n be the average number of adjacent nodes of every node in a HIN, the time cost for obtaining multi-hop relation is approximately n^l for each given starting node. Thus, leveraging such multi-hop relations of HINs may severely cause exponential time complexity and scalability issues [12, 17]. To alleviate such inefficiency, we propose a scalable way to extract meta-path based features to profile each user and item. Depending on the meta-paths used in this method, different types of meta-path features can be extracted. By using meta-paths involving a visual factor node type, the extracted meta-path features can be used to facilitate visually-aware recommendations subsequently.

The third component is an **explainable recommendation generation** method. Meta-paths are capable of extracting meaningful multi-hop relations [3]. Due to this strength, they have been tremendously used to improve the explainability of recommendations [18]. This paper introduces the concept of *meta-path based explainability* stemming from the proposed meta-path features. It allows us to quantify the "explainability" scores between user-item pairs based on a set of meta-paths. These scores can be leveraged in various recommender systems to provide explainability to these systems. However, since deep learning based recommender systems usually suffer from scalability and also interpretability/explainability issues [1], we propose a shallow recommendation model that jointly considers the proposed meta-path features and the explainability factor to produce explainable visually-aware recommendations. Moreover, compared with deep learning based recommender systems, our model requires less computational time and is more scalable.

Overall, unlike the previous work on visually-aware recommender systems, this work proposes a visually-aware recommender system that focuses on achieving both scalability and explainability. The contributions of this paper can be summarized as follows: (1) We propose a unified framework to bridge the gap of designing visually-aware recommender systems with high Scalability, Accuracy, and Explainability. (2) Instead of using visual information directly, we introduce visual factor nodes and visual relations to integrate visual factors into regular HINs and construct visually-augmented HINs. Also, we introduce probabilistic meta-paths to leverage multi-hop relations that dynamically involve both semantic and visual factors. (3) We propose a scalable meta-path feature extraction method to profile users and items with multi-hop relations efficiently. By using meta-paths that contain one or more visual factor node types, users' visual preferences can be modeled which subsequently facilitates visually-aware recommendation. (4) We introduce the concept of meta-path based explainability to quantify explainability between users and items. (5) We propose a shallow recommendation model that jointly considers meta-path features and the meta-path based explainability for efficiently generating explainable recommendations. (6) We conducted extensive experiments on real-world datasets for the Top-N recommendation task. We compared our approach with state-of-the-art approaches. The results were evaluated based on three metrics, i.e., Accuracy, Explainability, and Scalability. To the best of our knowledge, our work is the first that considers accuracy, scalability, and explainability aspects all in one visually-aware recommender system. Novel types of HIN and meta-path are proposed to effectively model users' visual preferences. A novel and efficient method for extracting meta-path features is proposed in this work. Also, the new concept of meta-path based explainability is introduced. All of these allow us to develop an effective and efficient explainable visually-aware recommender system.

The rest of the paper is organized as follows. In Section 2, the related work will be reviewed. In Section 4, the proposed SEV-RS will be discussed. The definitions of a visually-augmented HIN and a probabilistic meta-path will be first given. Then, we will explain how to efficiently generate user and item meta-path features based on visually-augmented HIN and probabilistic meta-paths. Based on these features, we will introduce the novel concept of meta-path based explainability and how to compute the explainability scores of user-item pairs. After that, the proposed approach for generating explainable recommendations will be discussed. Next, in Section 5, the experiments and results will be provided. Finally, the conclusions will be given in Section 6.

2. Related Work

Several recommender systems have been developed during the past decade. Collaborative Filtering (CF) [1] is one of the popular approaches that use either explicit (e.g. ratings) or implicit feedback (e.g. buy, tag and watch) from users to identify similar users and make recommendations based on users' similarities. Many models based on the CF approach have been proposed, e.g., CF-KNN that uses the K-Nearest-Neighbor method (KNN) for measuring similarity between users [19]. Although the CF approach is normally effective, its performance becomes poorer when user-item interactions are sparse [19]. To cope with the sparsity problem, reduction techniques such as Matrix Factorization (MF) [20] have been used to decompose a user-item interaction matrix into low-dimensional user/item latent factors. Based on the MF model, BPR-MF [21] was proposed to combine MF with Bayesian Personalized Ranking (BPR) scheme to learn user/item latent factors by using users' positive and negative items. The idea is to find low-dimensional user/item latent factors that can differentiate between positive and negative items of each user.

Numerous studies have shown the possibility of using additional information to improve recommendation accuracy. This includes visual information from item images in visually-aware recommender systems [22, 23, 7]. Owing to advances in computer vision and image processing, item image features can be extracted by using several feature extraction methods [24, 25, 26] or deep learning models such as convolutional neural networks (CNNs) [27]. These features have been proven to be highly useful for representing visual information of item images. They can be integrated into recommender systems as additional information along with user-item interactions to learn users' preferences more effectively. One of the firstly proposed visually-aware recommender systems is VBPR [22], a modified BPR-MF model that incorporates visual information into learning user/item latent factors. The new user/item latent factors subject to visual preferences were introduced and learned by projecting the extracted image features to the visual latent space using a learnable weight matrix. In [23], the weight matrix in VBPR was replaced by a deep learning module to better model more complex visual preferences. In [28], instead of directly using visual features learned from a deep learning model such as CNN to model users' visual preferences, the style features were proposed and used for the task of visual recommendation. These features were computed by subtracting the items' categorical representations from the visual features extracted from the CNN model. These studies have demonstrated the advantages of incorporating visual information to further improve recommendation accuracy.

HINs are networks/graphs containing connectivity information between objects represented by nodes. These nodes are connected by edges typically referred to as relations. Each node and relation in a HIN can be assigned with one or more types. HINs have been ubiquitously used to provide additional information in many recommender systems [3, 13, 14, 16]. Many proposed systems aim to leverage multi-hop information which can be obtained by several methods, e.g., Graph Convolutional Neural Network (GCNN) [29], RippleNet [15], and Graph Attention Network [16]. Apart from using these model architectures to capture structural information, various tools have been coupled with them to improve the performance, for instance, user-annotated tags in a Graph Attention Network model [30], sub-graphs extracted to capture high-level semantic information [31] and heterogeneous multi-graphs providing multiple relationships between two nodes [32]. Meta-paths are also another tool widely used for leveraging multi-hop information. Recently, meta-path based approaches have been popularly used to make recommendations in HINs [3]. The meta-path similarity measure framework of HINs provides a powerful mechanism for a user to measure the possibility of an unobserved user-item interaction in the network under different semantic assumptions. For example, metapath2vec [13] was proposed to generate random walks based on meta-paths and learn node representations by using the Skip-gram model. These node representations comprise multi-hop information and can be utilized in many recommendation models [33, 34, 17, 12]. In [14], the representations of users, items, and the aggregated meta-paths were modeled from path instances connecting the user with the item. In [35], path instances based on different meta-paths were used to attentively generate meta-path based context for learning user/item representations.

Both HINs and images have been proven to be useful for recommendations. Some models have been proposed to jointly leverage both of them. In our previous work [34], we introduced visually-augmented HINs where visual information from item images was integrated into HINs. We explored various image features to construct visually-augmented HINs and applied these HINs in recommender systems. To build recommender systems, metapath2vec [13] was adopted to learn node embeddings of these HINs. These embeddings then were used in the CF-KNN models to learn recommendations. The experimental results have shown that including visual factors in HINs and utilizing them via meta-paths improved the recommendation performance of CF-KNN models. Nonetheless, accuracy is no longer the only objective for modern recommender systems. This leads to a challenge in developing visually-aware recommender systems that also perform well in other aspects along with accuracy. In this paper, we extend our previous work to address three performance aspects, i.e., accuracy, explainability, and scalability. Previously, multi-hop relations in visually-augmented HINs were only used to improve recommendation accuracy in the shallow recommendation model. In this work, we develop a novel scalable method to efficiently leverage multi-hop relations in visually-augmented HINs. To improve recommendation explainability, we introduce a novel explainability definition based on multi-hop relations in visually-augmented HINs.

Accuracy has been a major focus in recommender system development. Many recent HIN-based recommender systems have shown their performances in terms of accuracy by using some deep learning models such as Long Short-Term Memory network [36] and Reinforcement Learning framework [37, 38, 11]. Another state-of-the-art approach is Knowledge Graph Attention Network (KGAT) [16] that uses a GCNN model and an attention mechanism to attentively propagate multi-hop relations in a HIN. Since HINs typically contain a large number of nodes and relations, it is challenging to develop HIN-based recommender systems that are scalable [11, 17, 12]. One approach to cope with this problem is to reduce the size of HINs by sampling a subset of paths [39, 29] or sub-graphs [40] instead of using all paths or an entire HIN. Another approach is to develop scalable learning techniques such as simplifying the GCNN models [41, 42] and pre-computing linear diffusion operations for efficient learning in Graph Neural Networks (GNNs) [43]. However, even with sampling or simplifying techniques, these systems can still suffer from scalability issues due to their structures and the large number of hyper-parameters.

Recently, explainability of recommender systems is required to increase the persuasiveness of recommendations, ensure users' trust, and support system maintenance and modification [44]. Many studies have explored how to constrain the systems to produce explainable recommendations rather than non-explainable ones. Some approaches modified the traditional shallow recommendation models such as the MF model [45, 46] and the BPR-MF model [33]. In these approaches, the explainability scores of user-item pairs were considered as an additional soft constraint. These scores were often defined by using user/item neighborhoods [45] or association rules [46, 33]. Such definitions focus on only hop-1 relations (e.g., user-item interactions) and ignore rich information from multi-hop relations. Some attempts on using multi-hop relations to improve the explainability have been made [15, 16, 47, 48]. However, using multi-hop relations to improve the explainability may result in a scalability issue. These requirements in the real-world situations have emphasized the importance of developing recommender systems capable of more than accurately predicting recommendations. Thus, how to design visually-aware recommender systems based on HINs with high accuracy, scalability, and explainability still needs to be explored.

3. Preliminaries

In this section, the definitions of HIN and meta-path are discussed. All notations used in this work are summarized in Table 1.

Definition 1. ***(HIN schema)*** *[10] Let* $\mathbb{G} = (\mathbb{N}, \mathbb{R}, \mathbb{W})$ *denote a HIN schema consists of a set of node types* $\mathbb{N}$*, a set of relation types* $\mathbb{R}$ *and a non-negative weight function* $\mathbb{W} : \mathbb{R} \to \mathfrak{R}$ *that maps each relation type to a non-negative real value in* $\mathfrak{R}$*. Let* $N_i, N_j \in \mathbb{N}$ *be any two node types,* $R_{N_i,N_j} \in \mathbb{R}$ *denotes the relation type connecting from* N_i *to* N_j*. For any* $R_{N_i,N_j} \in \mathbb{R}$*, let* $R^{-1}_{N_i,N_j}$ *denote an inverse relation type from* N_j *to* N_i*.*

Definition 2. ***(HIN)*** *[10] Given a HIN schema $\mathbb{G} = (\mathbb{N}, \mathbb{R}, \mathbb{W})$, a HIN is defined as a weighted and directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{R})$ where $\mathcal{N}$ is a set of nodes and $\mathcal{R}$ is a set of relations. Each node and relation is associated with their type mapping function: $\phi : \mathcal{N} \to \mathbb{N}$ and $\psi : \mathcal{R} \to \mathbb{R}$ respectively. Given nodes $x, y \in \mathcal{N}$, $r_{x,y}$ denotes a relation from x to y and its weight is denoted by $w(x, y) = \mathbb{W}(\psi(r_{x,y}))$.*

To leverage HINs for learning recommendations, meta-path based approaches have been widely used to access high-order connections between nodes. Unlike other approaches, they provide semantic meaning in multi-hop relations. Based on node and relation types in a HIN, a meta-path can be defined as follows:

Definition 3. ***(Meta-Path)*** *[49] Given a HIN $\mathcal{G}$, a meta-path m is defined as $N_1 \xrightarrow{R_{N_1,N_2}} N_2 \cdots N_l \xrightarrow{R_{N_l,N_{l+1}}} N_{l+1}$ (abbreviated as $N_1N_2 \cdots N_{l+1}$), describes a composite relation $R_{N_1,N_2} \circ \cdots \circ R_{N_l,N_{l+1}}$ between N_1 and N_{l+1} where $\circ$ denotes the composition operator on relations. A path $z = (n_1 n_2 \cdots n_{l+1})$ in $\mathcal{G}$ is called a **path instance** of m, if each n_i belongs to type N_i in m for all $i = 1, 2, ..., l+1$.*

4. Scalable and Explainable Visually-Aware Recommender System (SEV-RS)

In this section, we discuss the proposed SEV-RS framework. The goal of this framework is to learn visually-aware recommendations for achieving three performance aspects, i.e., accuracy, scalability, and explainability. SEV-RS consists of three components. Firstly, we discuss the first component which is a visually-augmented HIN. How to construct this augmented HIN and the proposed probabilistic meta-paths are described in this part. Then, for the second component, we discuss how to efficiently extract meta-path features by using the proposed scalable meta-path feature extraction method. In the third component, based on the meta-path features, we discuss how to generate explainable visually-aware recommendations.

4.1. Visually-Augmented HIN

A regular HIN typically contains only semantic factors (e.g., category, brand, etc.). To leverage visual information, this work proposes a visually-augmented

HIN that contains pivotal visual factors from item images. Based on this HIN, we can then use multi-hop relations to better profile users' visual preferences. We propose an approach of augmenting visual factors in a HIN to construct a visually-augmented HIN defined as follows:

Definition 4. ***(Visually-augmented HIN schema)*** *Given a HIN schema* $\mathbb{G}$*, a visually-augmented HIN schema is defined as* $\mathbb{G}' = (\mathbb{N}', \mathbb{R}', \mathbb{W}')$ *where* $\mathbb{N}' = \mathbb{N} \cup \{V\}$*,* $\mathbb{R}' = \mathbb{R} \cup \{R_{UV}, R_{PV}\}$*,* $\mathbb{W}' : \mathbb{R}' \to \mathfrak{R}$ *where* V *is a visual factor node type and* R_{UV} *and* R_{PV} *are visual relation types connecting a user node to a visual factor node and an item node to a visual factor node respectively.*

Example 1. ***(Visually-Augmented HIN Schema)*** Figure 1 shows an example of a visually-augmented HIN schema in a clothing domain. In this figure, all nodes and edges shown in solid lines are all semantic types. They belong to the original HIN schema. A visual factor node type (V) is added to the original schema along with two new relation types, i.e., user-visual factor relation (R_{UV} and R_{UV}^{-1}) and item-visual factor relation (R_{PV} and R_{PV}^{-1}). These additional nodes and edges are presented with dash lines in this figure.

Definition 5. ***(Visually-Augmented HIN)*** *Let* $\mathcal{V}$ *denote a set of visual factor nodes and* $\mathcal{R}_{\mathcal{V}}$ *denote a set of relations connecting semantic and visual factor nodes. A visually-augmented HIN* $\mathcal{G}' = (\mathcal{N}', \mathcal{R}', \mathcal{W}')$ *is a HIN with a schema* $\mathbb{G}'$ *where* $\mathcal{N}' = \mathcal{N} \cup \mathcal{V}$*,* $\mathcal{R}' = \mathcal{R} \cup \mathcal{R}_{\mathcal{V}}$ *and* $\mathbb{W}' : \mathbb{R}' \to \mathfrak{R}$ *denotes a non-negative weight function that maps each relation type to a real value in* $\mathfrak{R}$*.*

In order to construct a visually-augmented HIN, visual factor nodes must be first generated. Visual factor nodes are representatives of significant image features extracted from item images. These image features can be of any type such as local keypoint descriptors from SIFT [24], SURF [25] or ORB [26], color histograms or hidden layer outputs from pre-trained deep learning models. In this work, the features extracted from a hidden layer of the pre-trained CNN model are selected. They are referred to as CNN features in this paper. The features are extracted from the second fully-connected layer (i.e. FC7) of the Caffe reference model [27]. We select this feature type since it has been used in many applications including visually-aware recommendations [22]. Also, unlike other feature types that can capture only one type of image characteristics, this feature type can capture multiple characteristics such as texture, shape, and color, from the model. Visual factors are defined as cluster centers of the extracted image features. In this work, we use the k-mean clustering method to divide the extracted features into k_V clusters. Thus, we have k_V visual factors namely $\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_{k_V}$. Note that each of them is a vector that is a representative of image features (which are also vectors) within its cluster. Based on these visual factors, a set of visual factor nodes $\mathcal{V} = \{v_1, v_2, v_3, ..., v_{k_V}\}$ is formed and added to $\mathcal{N}'$ where v_i is a visual factor node of visual factor $\mathbf{v}_i$ for every $i = 1, 2, 3, ..., k_V$. After visual factors are generated, each item node is then connected to visual factor nodes depending on its image features. Specifically, for each image feature extracted from item p image, if it belongs to a cluster of a visual factor $\mathbf{v}_i$, then two new relations with the types R_{PV} and R_{PV}^{-1} between p and v_i are added to $\mathcal{R}'$. As for connecting user nodes to visual factor nodes, the user visual preference profile $\mathbf{v}^u$ must be computed first. Given a user u, let $\mathcal{P}_u$ be the set of all u's items. $\mathbf{v}^u$ is computed by applying mean pooling [50] on all visual factors of his/her items in $\mathcal{P}_u$. Then, $\mathbf{v}^u$ is compared with every visual factor. The cosine similarity is used as a metric in this case. The most similar k_s visual factors are selected. Then, the relations with the types R_{UV} and R_{UV}^{-1} between u and each of the selected visual factors are added to $\mathcal{R}'$.

Example 2. ***(Visually-Augmented HIN)*** Figure 1 shows a visually-augmented HIN with the schema defined in Example 1. The above plane presents a regular HIN with only semantic factors, i.e., category and brand. On the below plane, there are two visual factor nodes connected to user and item nodes via user-visual factor relations and item-visual factor relations.

4.1.1. Probabilistic Meta-Path

Regular meta-paths can represent only multi-hop relations that are static. For example, let U, P and C denote the user, item, and category node types. A meta-path $UPCP$ suggests that a user may like an item only because it is in the same category as a user's previously interacted item. In some cases, users' preferences may depend on a mixture of factors. For instance, a user may prefer an item in the same category or has a similar appearance as one of his/her items. We call such combinations of multiple factors *hybrid factors*.

To capture such preferences based on hybrid factors, we use a meta-path called *probabilistic meta-path* in which hybrid factors are considered based on pre-defined probabilities. It is defined as follows:

Definition 6 (Probabilistic Meta-Path). *Given a visually-augmented HIN* $\mathcal{G}'$*, a probabilistic meta-path* $m' = N_1 N_2 \cdots N_{i-1}\{\delta * N_i \oplus (1-\delta) * N_j\} N_{i+1} \cdots N_l$ *is defined as a sequence of node types, relation types, and their transition probability in schema* $\mathbb{G}'$ *of* $\mathcal{G}'$ *where* $\oplus$ *is a symbol that represents the "or" relation of the semantic node type* N_i *and the visual factor node type* N_j *and* δ *is a probability* $0 \le \delta \le 1$*. It contains at least one visual factor node type and one visual relation type. Starting from node type* N_{i-1}*, the next node type will go to semantic node type* N_i *with the probability* δ *and go to visual factor node type* N_j *with the probability* $1-\delta$*. For simplicity, we ignore the probability in the annotation and use* $N_1 N_2 \cdots N_{i-1}\{N_i \oplus N_j\} N_{i+2} \cdots N_l$ *to denote* m'*. The probability* δ *can be freely adjusted. When* $\delta = 1$*, visual factor node types will not be considered. They then become regular meta-paths without hybrid factors involved. In other words, regular meta-paths can be considered as special cases of probabilistic meta-paths where the probability* $\delta = 1$*.*

Example 3. ***(Probabilistic Meta-Path)*** Figure 1 shows an example of using the regular meta-path $UPCP$ and the probabilistic meta-path $UP\{C \oplus V\}P$ with the probabilities of going to category node type (C) δ and visual factor node type (V) $1-\delta$. The numbers and symbols on the edges indicate the probabilities assigned on those edges. By following the regular meta-path $UPCP$, a path instance ("User 1","T-shirt A","Category: T-shirt","T-shirt B") can be found. This suggests that "User 1" may like "T-shirt B" because it is in the same category as "T-shirt A". On the other hand, by following $UP\{C \oplus V\}P$, another path instance, ("User 1","T-shirt A","v_1","T-shirt B") can be found. It shows that "User 1" may also like "T-shirt B" because it has the same visual factor (v_1) as "T-shirt A". We can see that $UP\{C \oplus V\}P$ can reveal "User 1"'s preference in a more complex way compared to the regular meta-path $UPCP$.

4.2. Scalable Meta-Path Feature Extraction

Meta-paths have been used to determine the similarity (connectivity strength) between nodes in a HIN. Let u be a user, p be an item, m be a meta-path, and

$\mathcal{Z}_m$ be a set of path instances of m connecting u and p. Let $s(u,p,m)$ denote the meta-path based connectivity strength of u and p following m. It can be calculated as the sum of the probabilities of path instances $z \in \mathcal{Z}_m$ [49]:

$$s(u,p,m) = \sum_{z \in \mathcal{Z}_m} Pr(z) \tag{1}$$

The higher the sum of the probabilities, the higher the connectivity strength. To achieve accurate recommendations, it is critical to find the most informative or predictive meta-paths. For user u, if we can find the predictive meta-paths that lead to his/her observed items, then it is more likely that these meta-paths will help find those unobserved items that he/she will be interested in. Intuitively, if the total connectivity strength between u and his/her observed items following m is high, then meta-path m is predictive/important for u. To measure the importance of a meta-path for a user, we introduce the concept of *User-MetaPath association.*

Definition 7 (User-MetaPath Association). *User-MetaPath association is the aggregated meta-path based connectivity strengths between* u *and his/her observed items following* m*. It is defined as* $a_{u,m} = \sum_{p \in \mathcal{P}_u} s(u,p,m)$ *where* $\mathcal{P}_u$ *is the set of observed items of* u *and* $s(u,p,m)$ *is the meta-path based connectivity strength between user* u *and item* p *following meta-path* m*.*

Similarly, we can measure the importance of a meta-path for an item. For an item p, if the total connectivity strength between p and its observed users denoted as $\mathcal{U}_p$ following meta-path m is high, then m is important to p. We define the concept of *Item-MetaPath association* as follows.

Definition 8 (Item-MetaPath Association). *Item-MetaPath association is the aggregated meta-path based connectivity strengths between* p *and its observed users following meta-path* m*. It is defined as* $a_{p,m} = \sum_{u \in \mathcal{U}_p} s(u,p,m)$ *where* $\mathcal{U}_p$ *is the set of users interacted with* p *and* $s(u,p,m)$ *is the meta-path based connectivity strength between user* u *and item* p *following meta-path* m*.*

Both User-MetaPath and Item-MetaPath associations can be computed from any meta-paths including probabilistic meta-paths. However, the connectivity strength is normally computed from a regular meta-path. Thus, this work

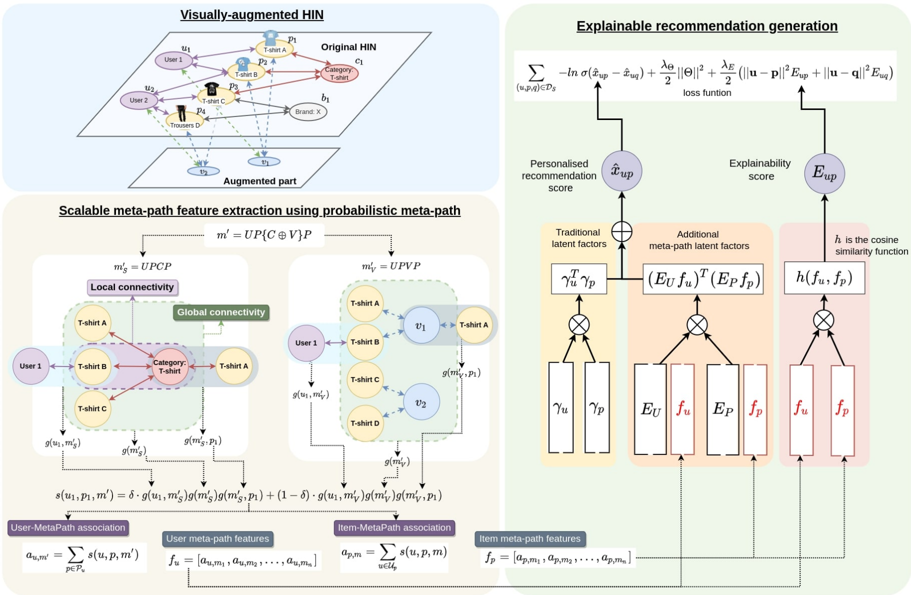


Figure 2: SEV-RS consisting of three parts: (1) a visually-augmented HIN, (2) scalable meta-path feature extraction and (3) explainable recommendation generation

proposes a novel method to compute the connectivity strength between user and item nodes based on a probabilistic meta-path. In this way, we are able to find the most informative or predictive probabilistic meta-path for each user/item.

Given a probabilistic meta-path $m' = N_1 N_2 \cdots N_{i-1}\{N_i \oplus N_j\}N_{i+2} \cdots N_l$, a path instance can follow either $m'_S = N_1 N_2 \cdots N_{i-1} N_i N_{i+2} \cdots N_l$ with the probability δ, or $m'_V = N_1 N_2 \cdots N_{i-1} N_j N_{i+2} \cdots N_l$ with the probability $1 - \delta$. Thus, we define the connectivity strength between user u and item p following m' as the weighted sum of connectivity strength following m'_S and m'_V as follows:

$$s(u, p, m') = \delta \cdot \sum_{z \in \mathcal{Z}_{m'_S}} Pr(z) + (1 - \delta) \cdot \sum_{z \in \mathcal{Z}_{m'_V}} Pr(z) \qquad (2)$$

where $\mathcal{Z}_{m'_S}$ and $\mathcal{Z}_{m'_V}$ are sets of path instances of m'_S and m'_V respectively.

Computing Eq. (2) requires all path instances in $\mathcal{Z}_{m'_S}$ and $\mathcal{Z}_{m'_V}$ which is not scalable. To address this issue, we propose a novel scalable approach to compute $s(u, p, m')$. Inspired by the processing of a sentence (i.e., a sequence of words) in Natural Language Processing, we apply the *first-order Markov assumption* to calculate $Pr(z)$. For any z, we assume that the probability of each node in z

depends only on its previous nodes. Thus, $Pr(z)$ can be computed by

$$Pr(z) = Pr(u, n_1, n_2, \cdots, n_l, p) = Pr(u)Pr(n_1|u)Pr(n_2|n_1)\cdots Pr(p|n_l) \quad (3)$$

where $Pr(u) = \frac{c_u}{|\mathcal{N}|}$ is the probability of node u in a HIN where c_u is the total number of user u nodes in a HIN and $|\mathcal{N}|$ is the total number of nodes in a HIN, and $Pr(y|x)$ is the probability of node y given x as a previous node in z for any $x, y \in \{u, n_1, n_2, ..., n_l, p\}$ computed by $Pr(y|x) = \frac{w(x,y)}{\sum_{n\in\mathcal{N}} w(x,n)}$ Note that, for every user u, $Pr(u)$ is constant since every user has only one node in a HIN (i.e., $c_u = 1$ for every user u) and the total number of nodes in a HIN is constant. Thus, we ignore this term in the connectivity computation.

Considering $\sum_{z\in\mathcal{Z}_{m'_S}}$ and $\sum_{z\in\mathcal{Z}_{m'_V}}$ in Eq. (2), they are equivalent to the summations over all possible combinations of node sequences following m'_S and m'_V respectively. Thus, these two summations can be replaced by the series of summations that consider all combinations of node sequences following m'_S and m'_V instead. Thus, we have

$$\begin{aligned}\sum_{z\in\mathcal{Z}_{m'_S}} Pr(z) &= \sum_{n_1\in\mathcal{N}_1}\sum_{n_2\in\mathcal{N}_2}\cdots\sum_{n_i\in\mathcal{N}_i}\cdots\sum_{n_l\in\mathcal{N}_l} Pr(n_1|u)Pr(n_2|n_1)\cdots Pr(p|n_l) \\ &= \sum_{n_1\in\mathcal{N}_1}\sum_{n_l\in\mathcal{N}_l} Pr(n_1|u)\sum_{n_2\in\mathcal{N}_2}\cdots\sum_{n_{l-1}\in\mathcal{N}_{l-1}} Pr(n_2|n_1)\cdots Pr(n_l|n_{l-1})Pr(p|n_l)\end{aligned} \quad (4)$$

and

$$\begin{aligned}\sum_{z\in\mathcal{Z}_{m'_V}} Pr(z) &= \sum_{n_1\in\mathcal{N}_1}\sum_{n_2\in\mathcal{N}_2}\cdots\sum_{n_j\in\mathcal{N}_j}\cdots\sum_{n_l\in\mathcal{N}_l} Pr(n_1|u)Pr(n_2|n_1)\cdots Pr(p|n_l) \\ &= \sum_{n_1\in\mathcal{N}_1}\sum_{n_l\in\mathcal{N}_l} Pr(n_1|u)\sum_{n_2\in\mathcal{N}_2}\cdots\sum_{n_{l-1}\in\mathcal{N}_{l-1}} Pr(n_2|n_1)\cdots Pr(n_l|n_{l-1})Pr(p|n_l)\end{aligned} \quad (5)$$

where $\mathcal{N}_k$ is the set of nodes of N_k type ($k = 1, 2, ..., l$). Both Eq. (4) and (5) can be computed similarly. Therefore, for simplicity, we first consider $\sum_{z\in\mathcal{Z}_{m'_S}} Pr(z)$ in Eq. (4). Let n be the average number of adjacent nodes per node, computing Eq. (4) requires time complexity of $O(n^l)$, which is computationally expensive. Therefore, we propose an alternative way to reduce the computational time by estimating the term $\sum_{n_2\in\mathcal{N}_2}\cdots\sum_{n_{l-1}\in\mathcal{N}_{l-1}} Pr(n_2|n_1)\cdots Pr(n_l|n_{l-1})$.

This term represents the connectivity from n_1 to n_l. This can be considered as *local connectivity* since it considers the relations between some particular nodes at the path-instance level. For example, in Figure 2, the purple curved dashed box represents the local connectivity between "T-shirt B" and "Category: T-shirt".

Computing the local connectivity from n_1 to n_l in each path instance is time-consuming. Instead of considering the connectivity between nodes at the path-instance level, we can use the connectivity between node types at the meta-path level to measure the importance of a meta-path. This connectivity is called *global connectivity* of a meta-path m denoted as $g(m)$. It is computed by

$$g(m) = \prod_{k=1}^{l-1} C(k, k+1), \tag{6}$$

where $C(k, k+1)$ denotes the probability of N_{k+1} type nodes given N_k type nodes computed by

$$C(k, k+1) = \frac{\sum_{n' \in \mathcal{N}_k} \sum_{n'' \in \mathcal{N}_{k+1}} w(n', n'')}{\sum_{n' \in \mathcal{N}_k} \sum_{n \in \mathcal{N}} w(n', n)} \tag{7}$$

where $w(n', n'')$ and $w(n', n)$ denote the weights of the relations from n' to n'' and from n' to n respectively. Each $C(k, k+1)$ indicates the connectivity between one node type to another node type. Considering them all, $g(m)$ therefore indicates the connectivity between general N_1 type nodes to N_l type nodes through $N_2, ..., N_{l-1}$. Without actual path instances, we use $g(m)$ to find how likely user u links to item p following meta-path m. This global connectivity is shown as the green curved dashed box in Figure 2. From this figure, we can see that the global connectivity measures the general connectivity between overall item nodes and overall category nodes, rather than the specific connectivity between one/some item nodes and one/some category nodes. After substituting the local connectivity with the global connectivity in Eq. (4), we have

$$\sum_{z \in \mathcal{Z}_{m'_S}} Pr(z) = g(u, m'_S) g(m'_S) g(m'_S, p) \tag{8}$$

where $g(u, m'_S) = \sum_{n_1 \in \mathcal{N}_1} Pr(n_1|u)$ and $g(m'_S, p) = \sum_{n_l \in \mathcal{N}_l} Pr(p|n_l)$. Similarly, $\sum_{z \in \mathcal{Z}_{m'_V}} Pr(z)$ in Eq. (5) is computed as follows:

$$\sum_{z \in \mathcal{Z}_{m'_V}} Pr(z) = g(u, m'_V) g(m'_V) g(m'_V, p) \tag{9}$$

Hence, $s(u, p, m')$ is computed as follows:

$$s(u, p, m') = \delta \cdot g(u, m'_S)g(m'_S)g(m'_S, p) + (1-\delta) \cdot g(u, m'_V)g(m'_V)g(m'_V, p) \quad (10)$$

Example 4 (User-MetaPath Association). Given the visually-augmented HIN in Figure 1, let all relations have the same weight $w(x, y) = w(y, x) = 1$. Let $\mathcal{P}$ denote the set of item nodes, $\mathcal{C}$ denote the set of category nodes and $\mathcal{B}$ denote the set of brand nodes. Given a probabilistic meta-path $m'_1 = UP\{C \oplus V\}P$ and $\delta = 0.4$, u_1's User-MetaPath association is computed by

$$a_{u_1, m'_1} = \sum_{p \in \mathcal{P}_{u_1}} s(u_1, p, m'_1) = s(u_1, p_1, m'_1) + s(u_1, p_2, m'_1) \approx 0.12.$$

Similarly, for $m'_2 = UP\{B \oplus V\}P$, we can calculate

$$a_{u_1, m'_2} = \sum_{p \in \mathcal{P}_{u_1}} s(u_1, p, m'_2) = s(u_1, p_1, m'_2) + s(u_1, p_2, m'_2) \approx 0.08.$$

(The complete details can be found in Appendix A). Since m'_1 has more weight than m'_2, thus, m'_1 is more important for "User 1" compared to m'_2.

Example 5 (Item-MetaPath Association). Given the same HIN shown in Figure 1 and the same probabilistic meta-path $m'_1 = UP\{C \oplus V\}P$ with $\delta = 0.4$, Item-MetaPath association between p_1 and m'_1, a_{p_1, m'_1}, is computed as follows:

$$a_{p_1, m'_1} = \sum_{u \in \mathcal{U}_{p_1}} s(u, p_1, m'_1) = s(u_1, p_1, m'_1) + s(u_2, p_1, m'_1) \approx 0.1$$

Similarly, for $m'_2 = UP\{B \oplus V\}P$, we can calculate

$$a_{p_1, m'_2} = \sum_{u \in \mathcal{U}_{p_1}} s(u, p_1, m'_2) = s(u_1, p_1, m'_2) + s(u_2, p_1, m'_2) \approx 0.07$$

(The complete details can be found in Appendix A). Since m'_1 has more weight than m'_2, thus, m'_1 is more important for "T-shirt A" compared to m'_2.

Usually, a group of meta-paths can better explain why a user is interested in an item than a single meta-path. We propose to use a group of meta-paths to generate user and item meta-path features. Given a set of meta-paths, multiple User-MetaPath and Item-MetaPath associations can be computed. Such associations of the same user/item can be used to form feature vectors of that user/item. Let $\mathcal{M} = \{m_1, m_2, ..., m_n\}$ be a set of meta-paths where

$m_1, m_2, ..., m_n$ are n pre-defined meta-paths. The *user meta-path feature* of u and the *item meta-path feature* of p are defined as $\mathbf{f}_u = [a_{u,m_1}, a_{u,m_2}, ..., a_{u,m_n}]$ and $\mathbf{f}_p = [a_{p,m_1}, a_{p,m_2}, ..., a_{p,m_n}]$ respectively. Both $\mathbf{f}_u$ and $\mathbf{f}_p$ enclose User-MetaPath and Item-MetaPath associations to represent a given user u and item p. Each dimension in $\mathbf{f}_u$ and $\mathbf{f}_p$ indicates how each meta-path in $\mathcal{M}$ is associated with user u and item p. This can be seen as profiling users/items based on their associations with different meta-paths. In terms of explainability, since each dimension in f_u and f_p is meaningful, we can leverage it to provide explainability in recommendations. If both $a_{u,m}$ and $a_{p,m}$ are high, then it can be assumed that m is mutually important for both u and p. In that case, m is potentially an explanation of why u prefers p, i.e., p is explainable for u based on m. Mathematically, we can use the dot product of $a_{u,m}$ and $a_{p,m}$ to reflect this assumption. Based on this assumption, this work introduces the concept of *meta-path based explainability* for quantifying the explainability between users and items based on multi-hop relations in a HIN.

Definition 9 (Meta-Path Based Explainability). *Given a user u, an item p, and a meta-path m, the meta-path based explainability between u and p is measured by the dot product of u's User-MetaPath association and p's Item-MetaPath association.*

From this definition, the higher u and p are associated with the same meta-path m, the higher the explainability between them. The same with existing approaches [45, 33], we can set up a threshold value τ. If the computed product is greater than τ, then item p is explainable for user u following meta-path m. Otherwise, item p is not explainable for user u following meta-path m.

4.3. Explainable Recommendation Generation

In this section, we describe how to generate explainable recommendations based on the meta-path features. We discuss the modified BPR-MF framework that leverages the meta-path features along with user-item interactions and how to integrate the explainability into the framework.

The BPR-MF framework is an effective and popularly used framework for learning recommendations. It ranks the candidate items based on the user-personalized recommendation scores. In the general BPR-MF, the recommendation score of a user u towards an item p denoted as $\hat{x}_{up}$ is computed by

$$\hat{x}_{up} = \alpha + \beta_u + \beta_p + \boldsymbol{\gamma_u}^T \boldsymbol{\gamma_p} \tag{11}$$

where α is a global offset, β_u and β_p are user and item bias terms, $\boldsymbol{\gamma_u}$ and $\boldsymbol{\gamma_p}$ are K_1-dimensional vectors of user u and item p latent factors respectively. The system is learned by using a Bayesian Personalized Ranking (BPR) framework leveraging positive and negative items in a dataset. For any user $u \in \mathcal{U}$, let $\mathcal{P}_u^+$ be a set of positive items of user u. A training sample set is defined as $\mathcal{D}_S = \{(u, p, q) | u \in \mathcal{U} \wedge p \in \mathcal{P}_u^+ \wedge q \in \mathcal{P} \setminus \mathcal{P}_u^+\}$ where p is a user's positive item and q is a user's negative item which is an unobserved item of a user u. A stochastic gradient-descent algorithm is adopted for training with a generic optimization criterion defined as follows:

$$\sum_{(u,p,q) \in \mathcal{D}_S} -ln\ \sigma(\hat{x}_{up} - \hat{x}_{uq}) + \lambda_\Theta ||\Theta||^2 \tag{12}$$

where $\hat{x}_{up}$ and $\hat{x}_{uq}$ are the recommendation scores of user u towards p and q respectively, σ is the sigmoid function and $||\Theta||^2$ is an L2 norm regularization term where λ_Θ is a regularization hyper-parameter and Θ denotes model parameters.

The traditional BPR-MF model involves only user-item interaction data for learning. In [22], the BPR-MF model was extended to incorporate visual information from item images. User and item latent visual factors were introduced to the traditional model. For each item, item latent visual factors are computed by projecting its image feature onto the visual latent space. Meanwhile, since there are no images of users, user latent visual factors in visual rating space are directly learned without projecting as item latent visual factors. Following this idea, the meta-path based features of both u and p can be integrated into the personalized recommendation score as follows:

$$\hat{x}_{up} = \alpha + \beta_u + \beta_p + \boldsymbol{\gamma_u}^T \boldsymbol{\gamma_p} + \boldsymbol{\theta_u}^T \boldsymbol{\theta_p} + \boldsymbol{\beta}_P^T \mathbf{f}_p + \boldsymbol{\beta}_U^T \mathbf{f}_u \tag{13}$$

where $\boldsymbol{\theta_u}$ and $\boldsymbol{\theta_p}$ are additional K_2-dimensional latent factors apart from the traditional latent factors $\boldsymbol{\gamma_u}$ and $\boldsymbol{\gamma_p}$, $\boldsymbol{\beta}_P$ is an item feature bias vector, and $\boldsymbol{\beta}_U$ is a user feature bias vector. These additional latent factors are called meta-path based latent factors since they are factorized based on the proposed user/item meta-path based features. They are computed by $\boldsymbol{\theta_u} = \mathbf{E}_U \mathbf{f}_u$ and $\boldsymbol{\theta_p} = \mathbf{E}_P \mathbf{f}_p$ where $\mathbf{E}_U$ and $\mathbf{E}_P$ are matrices projecting $\mathbf{f}_u$ and $\mathbf{f}_p$ into K_2-dimensional latent spaces respectively. Both $\mathbf{E}_U$ and $\mathbf{E}_P$ are additional parameters in this model. Overall, $\hat{x}_{up}$ is calculated from two parts, the traditional latent factors $\boldsymbol{\gamma_u}$ and $\boldsymbol{\gamma_p}$ (including their biases α, β_u and β_p) and the meta-path based latent factors $\boldsymbol{\theta_u}$ and $\boldsymbol{\theta_p}$ (including their biases $\boldsymbol{\beta}_P^T \mathbf{f}_p$ and $\boldsymbol{\beta}_U^T \mathbf{f}_u$). Unlike VBPR, our model also considers the feature from the user side to learn the additional latent factors of a user. Compared to most HIN-based models, it is also worth noting that our model can be used to incorporate multi-hop information from a set of meta-paths. In other words, instead of relying on a single meta-path, we can leverage multiple meta-paths altogether simultaneously. Also, any combination of meta-paths can be applied in this approach. This includes a combination of regular meta-paths, probabilistic meta-paths, and both.

Next, we introduce how to utilize the meta-path based features to increase the explainability of the proposed model. In [45], an explainable MF model which is a modification of the traditional MF model was proposed. This model jointly considers user-item interactions as in the traditional MF model and the explainability scores of user-item pairs as an additional soft constraint in the loss function. To measure the explainability between u and p based on a set of meta-paths $\mathcal{M}$, the explainability score E_{up} can be computed by the dot product of f_u and f_p. Since f_u and f_p are two vectors and can have significantly different vector magnitudes, we use the cosine similarity which is the normalized dot product of two vectors to compute $E_{up} = h(f_u, f_p)$ where $h(f_u, f_p)$ denotes the cosine similarity between f_u and f_p. Based on Definition (9), if p (or q) is explainable for u, then they should be close to each other in the latent space. Based on this assumption, the explainability scores are integrated into the loss function to constrain the distance between the user and item latent factors. The higher the explainability score, the closer both latent factors are. Thus, the original loss function in Eq. (12) is changed to

$$\sum_{(u,p,q)\in\mathcal{D}_S} -ln\ \sigma(\hat{x}_{up} - \hat{x}_{uq}) + \frac{\lambda_\Theta}{2}||\Theta||^2 + \frac{\lambda_E}{2}\left(||\mathbf{u}-\mathbf{p}||^2 E_{up} + ||\mathbf{u}-\mathbf{q}||^2 E_{uq}\right) \quad (14)$$

where $\mathbf{u} = [\boldsymbol{\gamma_u}; \boldsymbol{\theta_u}]$, $\mathbf{p} = [\boldsymbol{\gamma_p}; \boldsymbol{\theta_p}]$ and $\mathbf{q} = [\boldsymbol{\gamma_q}; \boldsymbol{\theta_q}]$ denote the final combined latent factors of u, p and q respectively, E_{up} and E_{uq} are the explainability scores and λ_E is a regularization hyper-parameter. If E_{up} is high, it will constrain $||\mathbf{u} - \mathbf{p}||$ to be lower to minimize the loss. Thus, $\mathbf{u}$ and $\mathbf{p}$ will be closer in the latent space. The same process applies for E_{uq} and the distance between $\mathbf{u}$ and $\mathbf{q}$. In this way, the meta-path features are used to constrain the recommender system to make the recommendations with high meta-path based explainability instead of any recommendations. Thus, given a set of meta-paths used for feature extraction, the recommendations made based on the extracted features can be explained by the meanings of these meta-paths. The proposed framework utilizing the meta-path features and the meta-path based explainability scores is illustrated in Figure 2. It is worth noting that our proposed framework uses a set of pre-defined meta-paths to constrain the explainability of recommendations. This is different from the previous work attempting to extract explanations along with predictions. For instance, in [47], meta-paths were not used during the learning process but were extracted as explanations along with the outputs. Also, compared to existing studies on using pre-defined meta-paths to improve the explainability, our framework addresses the issue of scalability and is more flexible. For example, compared to [48], meta-paths used in our framework are not limited to only symmetric meta-paths of length 3.

4.3.1. Complexity Analysis

In the proposed approach, the meta-path features f_u and f_p are computed as part of pre-processing. Given a meta-path $m = UN_1N_2 \cdots N_lP$, let n be the average number of adjacent nodes per node. Computing $s(u, p, m)$ by considering all possible path instances requires $O(n^l)$. This is more computationally expensive than the proposed method. From Eq. (6), computing $g(m)$ needs $O((l-1)n^2)$. Meanwhile, computing $\sum_{n_1 \in \mathcal{N}_1} Pr(n_1|u)$ and $\sum_{n_l \in \mathcal{N}_l} Pr(p|n_l)$ requires $O(n)$. In total, for any pair of a user/item and a meta-path, computing $s(u, p, m)$ requires $O((l-1)n^2) + O(n)$. Furthermore, $g(m)$ only depends on a meta-path. It can be pre-calculated once and used for all users/items. As a result, our method is more scalable compared to the method that uses actual path instances. As for explainable recommendation generation, the modified BPR-MF framework consists of the traditional part and the additional part as previously discussed. The first part requires $O(K_1)$ to update the user and item latent factors for each iteration. For the additional part, updating $\mathbf{E}_U$ and $\mathbf{E}_P$

needs $O(K_2|\mathcal{M}|)$. Updating $\boldsymbol{\beta}_U$ and $\boldsymbol{\beta}_P$ needs $O(K_2)$. Therefore, the proposed learning framework requires $O(K_2|\mathcal{M}|) + O(K_2)$, in addition to the traditional part of the BPR-MF model. This is scalable since the size of meta-path set $|\mathcal{M}|$ and the sizes of latent factors K_1 and K_2 are usually small.

5. Experiments

Experiments were conducted to answer the following research questions: RQ1: How does the proposed approach using the meta-path features perform compared with the baselines? RQ2: How does the proposed approach perform when it is applied to a visually-augmented HIN compared with the baselines? RQ3: How does the proposed approach perform when the meta-path based explainability is included compared with the baselines? and RQ4: Is the proposed approach scalable compared with the baselines?

5.1. Experimental Setup

The experiments were conducted on two real-world datasets:

- **MovieLens dataset**[1] [51], an extension of the MovieLens dataset called HetRec2011-MovieLens-2K. It contains user tagging data, movie genres, actors, directors, and tags. As for visual information, we used movie posters as image data in this work. These movie posters were scraped from the OMDB[2] website and matched with the movie titles in the dataset.

- **Amazon dataset**[3] [52], consisting of users' reviews and item metadata. The original dataset contains such user and item data in multiple categories. However, we only selected the "Clothing" subset for the experiments in this work. We only retained 5-rated reviews in the dataset to ensure the users' satisfaction for learning their preferences. The ratings are converted to implicit feedback to be used in our proposed model. For each item, a link to its image is provided in the dataset. We downloaded all item images from these links to use in our approach.

[1] https://grouplens.org, http://www.rottentomatoes.com, http://www.imdb.com
[2] http://www.omdbapi.com
[3] http://jmcauley.ucsd.edu/data/amazon/

Dataset	Node type	#nodes	Relation type	#relations
MovieLens	user (U)	1,132	R_{UP}	20,255
	item (P)	3,767	R_{PG}	8,861
	genre (G)	19	R_{PA}	97,791
	actor (A)	53,472	R_{PD}	3,756
	director (D)	1,672	R_{PT}	43,265
	tag (T)	5,209	R_{UV}	1,126
	visual factor (V)	100	R_{PV}	3,121
Amazon	user (U)	39,387	R_{UP}	214,696
	item (P)	23,030	R_{PC}	154,833
	category (C)	1,193	R_{PB}	3,942
	brand (B)	1,181	R_{PH}	65,514
	bought together (H)	25,207	R_{UV}	39,387
	visual factor (V)	100	R_{PV}	23,033

Table 2: The statistics of MovieLens and Amazon datasets

For both datasets, we filtered out those users who have less than two items and those items that have been interacted with by less than two users. The basic statistics of the visually-augmented HINs of both datasets are shown in Table 2. The number of visual factors k_V is 100 (i.e., $k = 100$ in the k-means clustering method). The number of representative visual factors per user is 1 ($k_s = 1$). The meta-paths used for generating the meta-path based features for both datasets are selected from the literature [11, 34]. They are shown in Table 3 where the second column lists the regular meta-paths while the third column lists the probabilistic meta-paths. The sizes of user/item latent factors, K_1 and K_2, were set to 150. Therefore, the final latent factors, $\mathbf{u}$ and $\mathbf{p}$, are 300-dimensional. We set $\lambda_\Theta = 5 \times 10^{-5}$ for both datasets. All experiments were conducted on a machine with dual-core Intel(R) 1.80GHz CPU, NVIDIA 16GB GPU, and 128GB RAM.

The proposed approach was evaluated in the Top-N recommendation task. Three evaluation aspects, i.e., Accuracy, Explainability, and Scalability were considered. As for Accuracy, it was evaluated by two commonly used metrics: *Mean Average Precision* (MAP@N) and *Mean Recall* (Recall@N) with $N = 1, 5, 10, 50, 100$. To evaluate Explainability, we adopted two metrics, i.e., *Mean Explainability Precision@N* (EP@N) and *Mean Explainability Recall@N* (ER@N) [45] defined as follows: $EP@N = \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\frac{|\mathcal{E}_u \bigcap \mathcal{Y}_u|}{|\mathcal{Y}_u|}$ and

Dataset	Meta-paths	Probabilistic meta-paths
MovieLens	$UPUP$, $UPUPUP$, $UPGP$, $UPGPUP$, $UPAP$, $UPAPUP$, $UPDP$, $UPDPUP$, $UPTP$, $UPTPTP$	$UP\{U\oplus V\}P$, $UP\{U\oplus V\}P\{U\oplus V\}$, $UP\{G\oplus V\}P$, $UP\{G\oplus V\}P\{U\oplus V\}$, $UP\{A\oplus V\}P$, $UP\{A\oplus V\}P\{U\oplus V\}$, $UP\{D\oplus V\}P$, $UP\{D\oplus V\}P\{U\oplus V\}$, $UP\{T\oplus V\}P$, $UP\{T\oplus V\}P\{T\oplus V\}$
Amazon	$UPUP$, $UPUPUP$, $UPCP$, $UPCPUP$, $UPBP$, $UPBPUP$, $UPHP$, $UPHPHP$	$UP\{U\oplus V\}P$, $UP\{U\oplus V\}P\{U\oplus V\}$, $UP\{C\oplus V\}P$, $UP\{C\oplus V\}P\{U\oplus V\}$, $UP\{B\oplus V\}P$, $UP\{B\oplus V\}P\{U\oplus V\}$, $UP\{H\oplus V\}P$, $UP\{H\oplus V\}P\{H\oplus V\}$

Table 3: The meta-paths used in the experiments on Amazon and MovieLens datasets

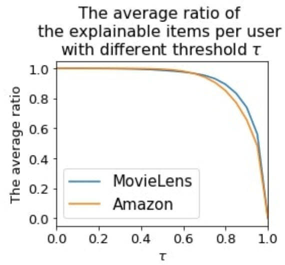


Figure 3: The average ratio of the explainable items per user with different τ

$ER@N = \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\frac{|\mathcal{E}_u \bigcap \mathcal{Y}_u|}{|\mathcal{E}_u|}$ where $\mathcal{U}$ denotes the users set, $\mathcal{E}_u$ denotes the set of explainable items of user u and $\mathcal{Y}_u$ denotes the set of Top-N recommended items of user u. For each user, the explainable items of that user are determined as in Definition 9, given the set of meta-paths defined in Table 3. Similarly to [45, 33], we can set up a threshold value τ to validate the explainable items of each user. Specifically, we say that p is explainable for u if $h(f_u, f_p) \geq \tau$ where τ is a pre-defined threshold. Figure 3 shows the average ratio of the explainable items to the user's items of each user in both datasets when τ is varied from 0 to 1. The ratio decreases as τ increases. To include most explainable items, we set $\tau = 0.55$ for both MovieLens dataset and Amazon dataset for evaluation. We selected $EP@5$ and $ER@5$ to evaluate the explainability performance.

5.2. Recommendation Accuracy Results (RQ1 and RQ2)

To answer RQ1 and RQ2, we compared two variations of our approach with the baselines as follows:

- **CF**: the CF-KNN model [19] that uses only user-item interactions.
- **BPR** [21]: the traditional BPR-MF model using user-item interactions.
- **VBPR** [22]: the modified BPR-MF model that jointly leverages user-item interactions and visual information. The same CNN features used in our approach were used as visual information in this model.
- **DVBPR** [23]: the modified version of VBPR that jointly trains the visual feature extraction model with the recommendation model instead of using the features from the pre-trained model.

- **DeepStyle** [28]: the BPR-MF model that incorporates the style features of items computed by subtracting item category representations from the visual features generated by CNN. We used the same CNN features as in VBPR for computing these style features.

- **MV** [34]: an approach using metapath2vec [13] with the CF-KNN model. We built multiple models of this approach based on each meta-paths in Table 3. The best model was selected for comparison.

- **GA** [16]: the state-of-the-art HIN-based model using Graph Attention Network. This model was applied to HINs without visual information.

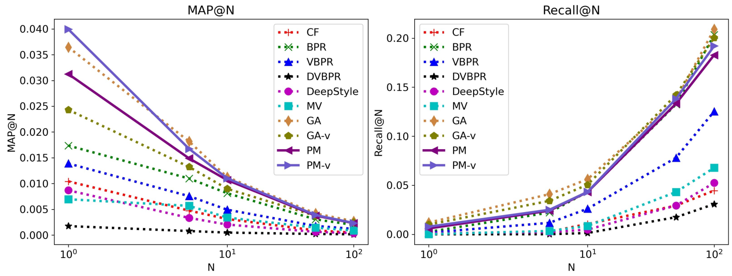
- **GA-v** [16]: the **GA** approach applied to visually-augmented HINs.

- **PM**: our model using regular meta-paths with regular HINs. Visual information and the meta-path based explainability were not considered.

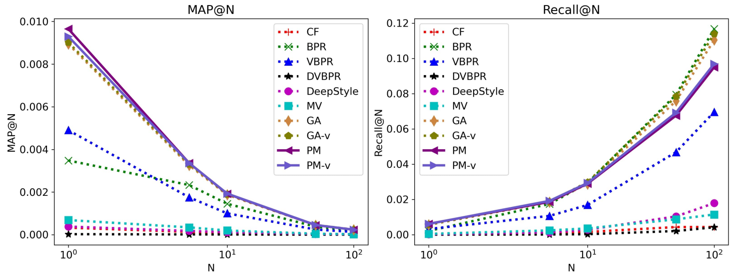
- **PM-v**: our proposed model using probabilistic meta-paths with visually-augmented HINs. The meta-path explainability was not considered. The parameter δ was varied among $\{0, 0.1, 0.2, ..., 1\}$ and the result with $\delta = 0.2$ were selected for comparison for both datasets.

For fair comparisons, the size of the final user/item latent factors or embeddings in **BPR**, **VBPR**, **DVBPR**, **DeepStyle**, **MV**, **GA** and **GA-v** were identically set to 300 as in our models. For **CF** and **MV**, the size of neighborhoods was set to 10. Other hyperparameter settings for the baselines were set as in their papers.

Figure 4 shows the results of both **MovieLens** and **Amazon** datasets. From these figures, we can see that **PM** outperformed **CF** in terms of both Precision and Recall. This can be explained that **CF** only uses single-hop relations (user-item interactions) for learning while the multi-hop relations are ignored in the model. Compared with **BPR**, **PM** outperformed it in terms of Precision on both datasets but performed similarly to **BPR** in terms of Recall. This shows the effectiveness of integrating multi-hop relations into the BPR-MF framework. **PM** also outperformed **MV** in terms of both Precision and Recall on both datasets. Although **MV** also utilizes meta-path based multi-hop relations, it can only consider a single meta-path at a time. On the other hand, our approach



(a) MovieLens



(b) Amazon

Figure 4: Accuracy comparison between the proposed approaches (without the explainability component) and other baselines

can consider multiple meta-paths simultaneously. Compared with the state-of-the-art deep learning model, **GA** outperformed **PM** on **MovieLens** dataset but they both performed similarly on **Amazon** dataset.

Next, we discuss how **PM-v** performed compared with the other models. As in Figure 4a, **PM-v** performed better than **VBPR**, **DVBPR** and **DeepStyle** in terms of both Precision and Recall. This shows that our model leveraged visual information to produce accurate recommendations more effectively than these visually-aware BPR-based models. Also, we can see that **PM-v** performed better than **PM** on **MovieLens** dataset. This suggests that the performance of our approach increased when using the visually-augmented HIN on this dataset. In fact, the performance of **PM-v** was enhanced up to the performance of **GA** which is a deep learning model. On the contrary, **GA-v** performed worse than **GA** in terms of both Precision and Recall on **MovieLens** dataset. This implies that the performance of the Graph Attention model dropped when it is applied to the visually-augmented HIN on this dataset. This result suggests that the Graph Attention model may not work well on the augmented HIN unlike our proposed approach **PM-v**. This demonstrates the effectiveness of our approach in leveraging visual information from a visually-augmented HIN. As for **Amazon** dataset, the results are shown in Figure 4b. From this figure, **GA**, **GA-v**, **PM**, and **PM-v** all performed similarly in terms of Precision. In terms of Recall, **PM** and **PM-v** performed slightly worse than **GA** and **GA-v**. One possible reason is that **Amazon** dataset contains numerous cold-start users which may limit the performances of these models.

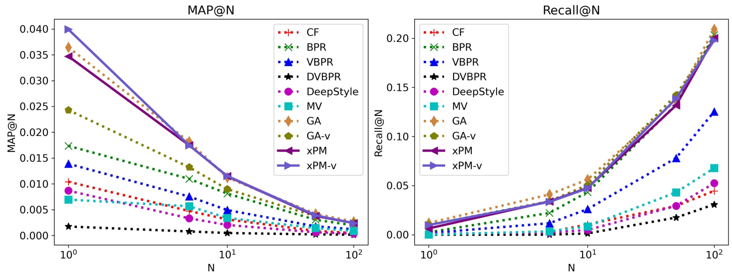
5.3. Explainability Discussion (RQ3)

In this part, we evaluated our approach involving the meta-path based explainability. We compared the same baselines as in the previous experiment with two variations of our approach:

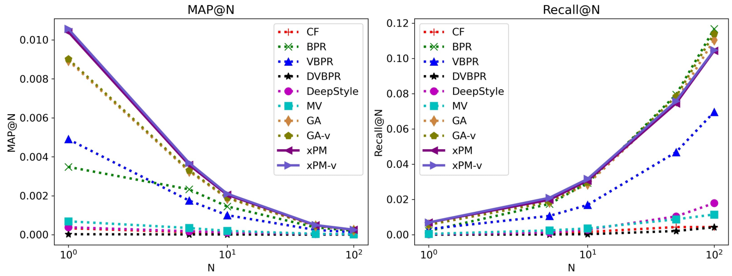
- **xPM**: our explainable model using regular meta-paths with regular HINs.
- **xPM-v**: our explainable model using probabilistic meta-paths with visually-augmented HINs.

We first examined the Accuracy performance of the proposed explainable approaches. Considering **MovieLens** dataset results in Figure 5a, **xPM** outperformed almost every baseline except **GA** in terms of Precision. In terms of Recall, **xPM** performed similarly to **GA** and **GA-v** while outperforming the others. Similar to the case of **PM** and **PM-v**, with the visually-augmented HIN, **xPM-v** performed better than **xPM** and even outperformed **GA** in terms of Precision. This depicts how our explainable approach can effectively utilize visual information for improving Accuracy in terms of Precision. As for **Amazon** dataset, the results are in Figure 5b. From this figure, both **xPM** and **xPM-v** have similar Precision and Recall. Both of them performed better than other baselines including **GA** and **GA-v** in terms of Precision. They also performed similarly to **GA** and **GA-v** in terms of Recall.

The Explainability results are shown in Figure 6. Considering the **MovieLens** dataset result in Figure 6a, **xPM** performed similarly to **GA** while it outperformed the other non-visually aware baselines. Considering **xPM-v**, its $EP@5$ is higher than most baselines except **DVBPR** and **DeepStyle**. This demonstrates that our approach produced more explainable items compared



(a) MovieLens



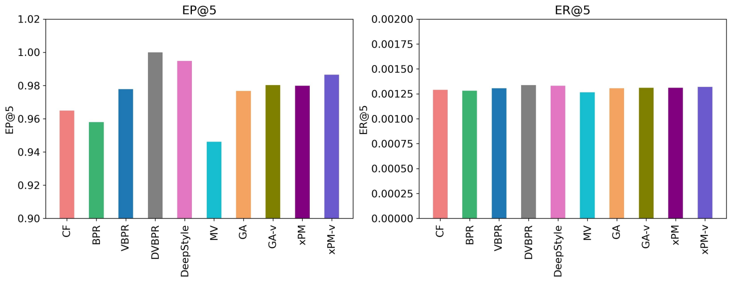
(b) Amazon

Figure 5: Accuracy comparison between the proposed approaches (with the explainability component) and other baselines

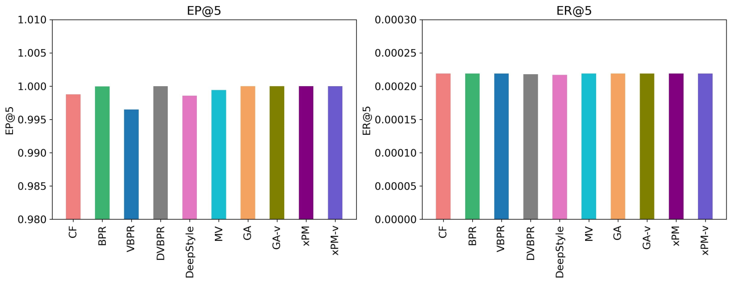
to most baselines, especially when utilizing visual information on this dataset. Although **DVBPR** and **DeepStyle** performed well regarding Explainability, they clearly performed worse than our approaches in terms of Precision and Recall. This suggests that **xPM-v** maintained a better trade-off between Accuracy and Explainability compared to **DeepStyle**. For **Amazon** dataset, Figure 6b shows that the $EP@5$ performances of all approaches are quite similar. One possible reason is that e-commerce purchase behaviors are easier to explain than movie preference rating/tagging behaviors. For both datasets, the $ER@5$ performances are similar for all the compared approaches.

5.4. Scalability Discussion (RQ4)

In this part, we discuss the Scalability of our approach **xPM-v** and the other two baselines that performed well in Accuracy, i.e., **BPR** and **GA**. We compared the computational time of these approaches applied on a synthetic dataset that includes 4 sub-datasets namely SD1, SD2, SD3, and SD4. Each sub-dataset contains a different number of relations at a different scale, i.e., 10^4, 10^5,



(a) EP@5 of MovieLens



(b) EP@5 of Amazon

Figure 6: Explainability comparison between the proposed approaches (with the explainability component) and other baselines

10^6, and 10^7. The statistics of these sub-datasets are shown in Table 4. For all models, the training batch size was set to 16 and they were trained for 10 epochs. The results are in Figure 7. From this figure, the computational time of **xPM-v** is slightly higher than **BPR** because **xPM-v** requires additional time for computing the meta-path features and the meta-path based explainability scores and updating the additional parameters in the modified BPR-MF framework. However, compared to **GA**, the computational time of **xPM-v** is much lower, especially for those large-scaled datasets (10^6 and 10^7). These results suggest that our proposed model **xPM-v** achieved close or similar performances with the popular shallow model **BPR**, from the aspect of Scalability. Also, it has significantly higher Scalability than the deep learning model **GA**.

To examine the Scalability of using probabilistic meta-paths and the meta-path based explainability, we compared the computational time of **PM**, **PM-v**, **xPM** and **xPM-v** as shown in Figure 7. From this figure, we can see that all of these variations had similar computational time with the differences for the sub-

Synthetic Dataset	Scale	#nodes	#relations
SD1	10^4	4,356	9,986
SD2	10^5	30,902	100,000
SD3	10^6	90,000	1,000,000
SD4	10^7	90,001	10,000,000

Table 4: The statistics of the synthetic datasets

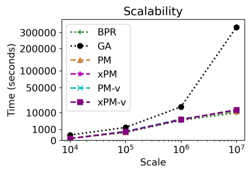


Figure 7: Scalability comparison between **xPM-v** and the baselines

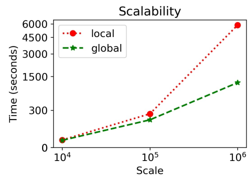
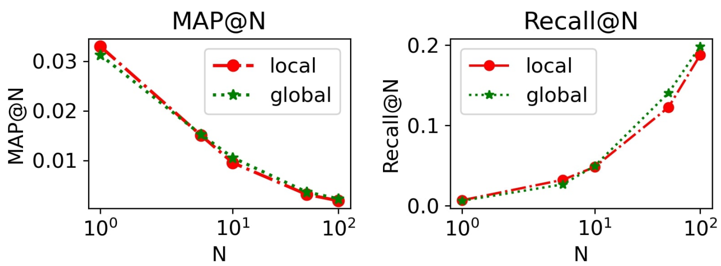


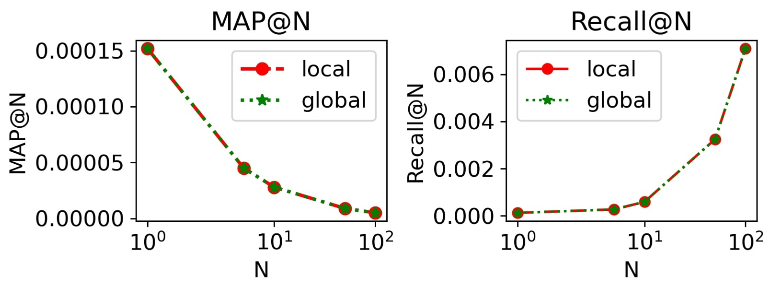
Figure 8: Scalability comparison between global and local connectivity

dataset SD4, the largest one. However, these differences are not as vast as the difference between **xPM-v** and **GA** discussed previously. Thus, using probabilistic meta-paths for computing the meta-path features and incorporating the meta-path based explainability can be considered scalable in this experiment.

We also compared Scalability and Accuracy of using the local connectivity in Eq. (4) and (5) and the global connectivity in Eq. (8) and (9). Given a set of meta-paths $\{UP, UPUP, UPUPUP\}$ that are only based on the user-item interaction, we consider the computational time that each method used for computing the meta-path features for the synthetic dataset. The results are shown in Figure 8. We can see that the proposed global connectivity method spent less computational time compared to the local connectivity method. This demonstrates the scalability of computing meta-path features using the global connectivity. Furthermore, we also examined the Accuracy of the proposed approach using the global connectivity and the local connectivity. This is to validate whether using the global connectivity in the proposed approach affects the Accuracy or not. For this experiment, we used **MovieLens** dataset and three meta-paths $\{UP, UPUP, UPUPUP\}$ for meta-path feature extraction. The Accuracy results of **xPM-v** based on the local connectivity and the global connectivity are shown in Figure 9. From this figure, both Precision and Recall of **xPM-v** using the local and global connectivity are similar. This suggests that using the proposed global connectivity in the proposed approach is as accurate as using the local connectivity, but more scalable.



(a) MovieLens



(b) Amazon

Figure 9: Accuracy of **xPM-v** using the local/global connectivity

6. Conclusions

This work proposed a scalable and explainable visually-aware recommender system called SEV-RS. It utilizes multi-hop relations for making visually-aware explainable recommendations. Specifically, we proposed a method to construct visually-augmented HINs. Within these HINs, probabilistic meta-paths were introduced to utilize the combinations of semantic and visual factors. Based on probabilistic meta-paths, a scalable way to extract meta-path based features to profile users and items was proposed. This method can efficiently leverage a set of meta-paths for the better use of multi-hop relations. To achieve the explainability, the concept of meta-path based explainability was introduced to quantify the explainability scores of user-item pairs based on a set of meta-paths. To generate explainable recommendations, we proposed a shallow BPR-based recommendation algorithm that integrates the proposed meta-path features and the explainability factor into the learning process.

SEV-RS was evaluated in the Top-N recommendation task based on three evaluation metrics, i.e., Accuracy, Explainability, and Scalability. Extensive experiments were conducted on two real-world datasets, i.e., HetRec2011-MovieLens-2K dataset and Amazon dataset in "Clothing" category, and one syntactic dataset. The results show that SEV-RS can produce more accurate recommendations according to the higher Precision and Recall compared with the baselines. We compared the performances of the Graph Attention Network model applied to visually-augmented HINs and our approach. The results show that SEV-RS can leverage visual information in visually-augmented HINs more effectively than the Graph Attention Network model. Also, the results show that SEV-RS can generate more explainable recommendations with higher Explainability Precision and Explainability Recall values. This indicates that our approach does not only recommend items of users' interests but also takes the explainability of items into consideration. As for Scalability, we conducted experiments on a synthetic dataset consisting of four sub-datasets with different scales. We compared the computational time of each model. The results show that SEV-RS can achieve similar scalability performances as the shallow BPR-MF model. It also required much less computational time compared to the Graph Attention Network model for large-scale sub-datasets. We also demonstrated that the additional computational time required for executing the explainability part in SEV-RS is trivial and thus this approach is scalable. Lastly, we compared the use of the proposed scalable meta-path feature extraction method and the straight-forwarding method in SEV-RS. The results show that using the proposed scalable method achieved similar Accuracy but cost significantly less computational time than using the straight-forwarding method.

As selecting meta-paths to produce more accurate and explainable recommendations can be difficult, for future work, we aim to overcome this problem by proposing a method to select or validate suitable meta-paths for ensuring accurate and explainable recommendations. This will further enhance the explainability and increase users' trust in the recommender systems.

References

[1] S. Zhang, L. Yao, A. Sun, Y. Tay, Deep learning based recommender system: A survey and new perspectives, ACM Computing Surveys 52 (1).

[2] J. Lu, D. Wu, M. Mao, W. Wang, G. Zhang, Recommender system application developments: A survey, Decision Support Systems 74 (2015) 12–32.

[3] Q. Guo, F. Zhuang, C. Qin, H. Zhu, X. Xie, H. Xiong, Q. He, A survey on knowledge graph-based recommender systems (2020).

[4] J. Bobadilla, F. Ortega, A. Hernando, A. Gutiérrez, Recommender systems survey, Knowledge-Based Systems 46 (2013) 109–132.

[5] R. He, C. Fang, Z. Wang, J. McAuley, Vista: A visually, socially, and temporally-aware model for artistic recommendation, in: Proceedings of the 10th ACM Conference on Recommender Systems, RecSys '16, Association for Computing Machinery, New York, NY, USA, 2016, p. 309–316.

[6] V. Jagadeesh, R. Piramuthu, A. Bhardwaj, W. Di, N. Sundaresan, Large scale visual recommendations from street fashion images, in: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, Association for Computing Machinery, New York, NY, USA, 2014, p. 1925–1934.

[7] X. Chen, H. Chen, H. Xu, Y. Zhang, Y. Cao, Z. Qin, H. Zha, Personalized fashion recommendation with visual explanations based on multimodal attention network: Towards visually explainable recommendation, Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (2019) 765–774.

[8] M. Hou, L. Wu, E. Chen, Z. Li, V. W. Zheng, Q. Liu, Explainable fashion recommendation: A semantic attribute region guided approach, in: IJCAI International Joint Conference on Artificial Intelligence, Vol. 2019-Augus, 2019, pp. 4681–4688.

[9] P. Liu, L. Zhang, J. A. Gulla, Dynamic attention-based explainable recommendation with textual and visual fusion, Information Processing & Management (2019) 102099.

[10] Y. Sun, Y. Yu, J. Han, Ranking-based clustering of heterogeneous information networks with star network schema, in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '09, Association for Computing Machinery, New York, NY, USA, 2009, p. 797–806.

[11] H. Liang, Drprofiling: Deep reinforcement user profiling for recommendations in heterogenous information networks, IEEE Transactions on Knowledge and Data Engineering (2020) 1–1.

[12] H. Liang, Z. Liu, T. Markchom, Relation-aware blocking for scalable recommendation systems, in: Proceedings of the 31st ACM International Conference on Information & Knowledge Management, CIKM '22, Association for Computing Machinery, New York, NY, USA, 2022, p. 4214–4218.

[13] Y. Dong, N. V. Chawla, A. Swami, metapath2vec: Scalable representation learning for heterogeneous networks, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017, pp. 135–144.

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

[15] H. Wang, F. Zhang, J. Wang, M. Zhao, W. Li, X. Xie, M. Guo, Ripplenet: Propagating user preferences on the knowledge graph for recommender systems, in: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 2018, p. 417–426.

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

[17] H. Liang, T. Markchom, Tne: A general time-aware network representation learning framework for temporal applications, Knowledge-Based Systems 240 (C).

[18] Y. Zhang, X. Chen, Explainable recommendation: A survey and new perspectives, Foundations and Trends® in Information Retrieval 14 (1) (2020) 1–101.

[19] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, IEEE Transactions on Knowledge and Data Engineering 17 (6) (2005) 734–749.

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

[21] S. Rendle, C. Freudenthaler, Z. Gantner, L. Schmidt-Thieme, BPR: Bayesian personalized ranking from implicit feedback, in: Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence, 2009.

[22] R. He, J. McAuley, VBPR: Visual bayesian personalized ranking from implicit feedback, in: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI'16, AAAI Press, 2016, p. 144–150.

[23] W.-C. Kang, C. Fang, Z. Wang, J. McAuley, Visually-aware fashion recommendation and design with generative image models, 2017 IEEE International Conference on Data Mining (ICDM) (2017) 207–216.

[24] D. Lowe, Object recognition from local scale-invariant features, in: Proceedings of the Seventh IEEE International Conference on Computer Vision, Vol. 2, 1999, pp. 1150–1157 vol.2.

[25] H. Bay, T. Tuytelaars, L. Van Gool, Surf: Speeded up robust features, in: Proceedings of the 9th European Conference on Computer Vision, Vol. 3951, 2006, pp. 404–417.

[26] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, ORB: An efficient alternative to sift or surf, 2011 International Conference on Computer Vision.

[27] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: Convolutional architecture for fast feature embedding, in: Proceedings of the 22nd ACM International Conference on Multimedia, 2014, p. 675–678.

[28] Q. Liu, S. Wu, 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, SIGIR '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 841–844.

[29] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, J. Leskovec, Graph convolutional neural networks for web-scale recommender systems, in: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining, 2018, pp. 974–983.

[30] G. Wang, H. Wang, J. Liu, Y. Yang, Leveraging the fine-grained user preferences with graph neural networks for recommendation, World Wide Web (2022) 1–23.

[31] X. Song, J. Li, Y. Tang, T. Zhao, Y. Chen, Z. Guan, Jkt: A joint graph convolutional network based deep knowledge tracing, Information Sciences 580 (2021) 510–523.

[32] Y. Li, Z. Fan, D. Yin, R. Jiang, J. Deng, X. Song, Hmgcl: Heterogeneous multigraph contrastive learning for lbsn friend recommendation, World Wide Web.

[33] M. G. Ozsoy, D. O'Reilly-Morgan, P. Symeonidis, E. Z. Tragos, N. Hurley, B. Smyth, A. Lawlor, MP4Rec: Explainable and accurate top-n recommendations in heterogeneous information networks, IEEE Access.

[34] T. Markchom, H. Liang, Augmenting visual information in knowledge graphs for recommendations, in: 26th International Conference on Intelligent User Interfaces, 2021, p. 475–479.

[35] H. Chen, Y. Li, X. Sun, G. Xu, H. Yin, Temporal meta-path guided explainable recommendation, Proceedings of the 14th ACM International Conference on Web Search and Data Mining.

[36] X. Wang, D. Wang, C. Xu, X. He, Y. Cao, T.-S. Chua, Explainable Reasoning over Knowledge Graphs for Recommendation, Proceedings of the AAAI Conference on Artificial Intelligence 33 (2019) 5329–5336.

[37] X. Wang, Y. Chen, J. Yang, L. Wu, Z. Wu, X. Xie, A reinforcement learning framework for explainable recommendation, in: 2018 IEEE International Conference on Data Mining (ICDM), 2018, pp. 587–596.

[38] Y. Xian, Z. Fu, S. Muthukrishnan, G. de Melo, Y. Zhang, Reinforcement knowledge graph reasoning for explainable recommendation, in: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019, p. 285–294.

[39] W. L. Hamilton, R. Ying, J. Leskovec, Inductive representation learning on large graphs, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, Curran Associates Inc., Red Hook, NY, USA, 2017, p. 1025–1035.

[40] Y. Feng, B. Hu, F. Lv, Q. Liu, Z. Zhang, W. Ou, ATBRG: Adaptive target-behavior relational graph network for effective recommendation, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2020.

[41] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, 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, 2020, p. 639–648.

[42] J. Zhao, Z. Zhou, Z. Guan, W. Zhao, W. Ning, G. Qiu, X. He, Intentgc: A scalable graph convolution framework fusing heterogeneous information for recommendation, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, p. 2347–2357.

[43] F. Frasca, E. Rossi, D. Eynard, B. Chamberlain, M. Bronstein, F. Monti, Sign: Scalable inception graph neural networks (2020).

[44] N. Tintarev, J. Masthoff, A survey of explanations in recommender systems, in: 2007 IEEE 23rd International Conference on Data Engineering Workshop, 2007, pp. 801–810.

[45] B. Abdollahi, O. Nasraoui, Using explainability for constrained matrix factorization, in: Proceedings of the Eleventh ACM Conference on Recommender Systems, 2017, p. 79–83.

[46] G. Peake, J. Wang, Explanation mining: Post hoc interpretability of latent factor models for recommendation systems, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, p. 2060–2069.

[47] Y. Yang, Z. Guan, J. Li, W. Zhao, J. Cui, Q. Wang, Interpretable and efficient heterogeneous graph convolutional network, IEEE Transactions on Knowledge and Data Engineering (2021) 1–1.

[48] M. Zhang, G. Wang, L. Ren, J. Li, K. Deng, B. Zhang, Metonr: A meta explanation triplet oriented news recommendation model, Knowledge-Based Systems 238 (C).

[49] Y. Sun, J. Han, X. Yan, P. S. Yu, T. Wu, Pathsim: Meta path-based top-k similarity search in heterogeneous information networks, Proceedings of the VLDB Endowment 4 (11) (2011) 992–1003.

[50] H. Liang, T. Baldwin, A probabilistic rating auto-encoder for personalized recommender systems, in: Proceedings of the 24th ACM International Conference on Information and Knowledge Management, 2015, pp. 1863–1866.

[51] I. Cantador, P. Brusilovsky, T. Kuflik, 2nd workshop on information heterogeneity and fusion in recommender systems (HetRec 2011), in: Proceedings of the 5th ACM conference on Recommender systems, 2011.

[52] R. He, J. McAuley, Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering, in: Proceedings of the 25th International Conference on World Wide Web, 2016, p. 507–517.

Appendix A. Examples of Computing User-MetaPath and Item-MetaPath Associations

Example 6 (User-MetaPath Association). Considering the visually-augmented HIN in Figure 1, suppose that all relations have the same weight 1, i.e., $w(x, y) = w(y, x) = 1$. Let $\mathcal{P}$ denote the set of item nodes, $\mathcal{C}$ denote the set of category nodes and $\mathcal{B}$ denote the set of brand nodes in this HIN. Given a probabilistic meta-path $m'_1 = UP\{C \oplus V\}P$ and $\delta = 0.4$, u_1's User-MetaPath association is computed by

$$a_{u_1,m'_1} = \sum_{p \in \mathcal{P}_{u_1}} s(u_1, p, m'_1) = s(u_1, p_1, m'_1) + s(u_1, p_2, m'_1) \tag{A.1}$$

where

$$s(u_1, p_1, m'_1) = \delta \cdot g(u_1, m'_{1S})g(m'_{1S})g(m'_{1S}, p_1) + (1 - \delta) \cdot g(u_1, m'_{1V})g(m'_{1V})g(m'_{1V}, p_1) \tag{A.2}$$

and

$$s(u_1, p_2, m'_1) = \delta \cdot g(u_1, m'_{1S})g(m'_{1S})g(m'_{1S}, p_2) + (1 - \delta) \cdot g(u_1, m'_{1V})g(m'_{1V})g(m'_{1V}, p_2) \tag{A.3}$$

where

$$g(u_1, m'_{1S}) = \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_1) \tag{A.4}$$

$$= Pr(p_1|u_1) + Pr(p_2|u_1) + Pr(p_3|u_1) + Pr(p_4|u_1) + Pr(p_5|u_1) \tag{A.5}$$

$$= 1/3 + 1/3 + 0/3 + 0/3 + 0/3 = 2/3, \tag{A.6}$$

$$\tag{A.7}$$

$$g(u_1, m'_{1V}) = \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_1) \tag{A.8}$$

$$= Pr(p_1|u_1) + Pr(p_2|u_1) + Pr(p_3|u_1) + Pr(p_4|u_1) + Pr(p_5|u_1) \tag{A.9}$$

$$= 1/3 + 1/3 + 0/3 + 0/3 + 0/3 = 2/3, \tag{A.10}$$

$$\tag{A.11}$$

$$g(m'_{1S}, p_1) = \sum_{n_2 \in \mathcal{C}} Pr(p_1|n_2) = Pr(p_1|c_1) = 1/3, \tag{A.12}$$

$$g(m'_{1V}, p_1) = \sum_{n_2 \in \mathcal{V}} Pr(p_1|n_2) = Pr(p_1|v_1) + Pr(p_1|v_2) = 1/3 + 0/3 = 1/3, \tag{A.13}$$

$$g(m'_{1S}, p_2) = \sum_{n_2 \in \mathcal{C}} Pr(p_2|n_2) = Pr(p_2|c_1) = 13, \tag{A.14}$$

$$g(m'_{1V}, p_2) = \sum_{n_2 \in \mathcal{V}} Pr(p_2|n_2) = Pr(p_2|v_1) + Pr(p_2|v_2) = 1/3 + 0/3 = 1/3, \tag{A.15}$$

$$g(m'_{1S}) = C(P, C) = 3/14 \tag{A.16}$$

$$g(m'_{1V}) = C(P, V) = 4/14, \tag{A.17}$$

where 3 is the total number of relations from P to C, 4 is the total number of relation from P to V and 14 is the total number of relations from P to any type including the additional visual relation type. Thus

$$s(u_1, p_1, m'_1) = 0.4 \cdot (2/3)(3/14)(1/3) + (0.6) \cdot (2/3)(4/14)(1/3) \approx 0.06 \tag{A.18}$$

and

$$s(u_1, p_2, m'_1) = 0.4 \cdot (2/3)(3/14)(1/3) + (0.6) \cdot (2/3)(4/14)(1/3) \approx 0.06 \tag{A.19}$$

$$a_{u_1,m'_1} = s(u_1, p_1, m'_1) + s(u_1, p_2, m'_1) \approx 0.12. \tag{A.20}$$

Similarly, for $m'_2 = UP\{B \oplus V\}P$, we can calculate

$$a_{u_1,m'_2} = \sum_{p \in \mathcal{P}_{u_1}} s(u_1, p, m'_2) = s(u_1, p_1, m'_2) + s(u_1, p_2, m'_2) \tag{A.21}$$

where

$$s(u_1, p_1, m'_2) = \delta \cdot g(u_1, m'_{2S})g(m'_{2S})g(m'_{2S}, p_1) + (1-\delta) \cdot g(u_1, m'_{2V})g(m'_{2V})g(m'_{2V}, p_1) \tag{A.22}$$

and

$$s(u_1, p_2, m'_2) = \delta \cdot g(u_1, m'_{2S})g(m'_{2S})g(m'_{2S}, p_2) + (1-\delta) \cdot g(u_1, m'_{2V})g(m'_{2V})g(m'_{2V}, p_2) \tag{A.23}$$

where

$$g(u_1, m'_{2S}) = \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_1) \tag{A.24}$$

$$= Pr(p_1|u_1) + Pr(p_2|u_1) + Pr(p_3|u_1) + Pr(p_4|u_1) + Pr(p_5|u_1) \tag{A.25}$$

$$= 1/3 + 1/3 + 0/3 + 0/3 + 0/3 = 2/3, \tag{A.26}$$

$$g(u_1, m'_{2V}) = \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_1) \tag{A.27}$$

$$= Pr(p_1|u_1) + Pr(p_2|u_1) + Pr(p_3|u_1) + Pr(p_4|u_1) + Pr(p_5|u_1) \tag{A.28}$$

$$= 1/3 + 1/3 + 0/3 + 0/3 + 0/3 = 2/3, \tag{A.29}$$

$$g(m'_{2S}, p_1) = \sum_{n_2 \in \mathcal{B}} Pr(p_1|n_2) = Pr(p_1|b_1) = 0, \tag{A.30}$$

$$g(m'_{2V}, p_1) = \sum_{n_2 \in \mathcal{V}} Pr(p_1|n_2) = Pr(p_1|v_1) + Pr(p_1|v_2) = 1/3 + 0/3 = 1/3, \tag{A.31}$$

$$g(m'_{2S}, p_2) = \sum_{n_2 \in \mathcal{B}} Pr(p_2|n_2) = Pr(p_2|b_1) = 0, \tag{A.32}$$

$$g(m'_{2V}, p_2) = \sum_{n_2 \in \mathcal{V}} Pr(p_2|n_2) = Pr(p_2|v_1) + Pr(p_2|v_2) = 1/3 + 0/3 = 1/3, \tag{A.33}$$

$$g(m'_{2S}) = C(P, B) = 2/14, \tag{A.34}$$

$$g(m'_{2V}) = C(P, V) = 4/14. \tag{A.35}$$

Thus,

$$s(u_1, p_1, m'_2) = 0.4 \cdot (2/3)(2/14)(0) + 0.6 \cdot (2/3)(4/14)(1/3) \approx 0.04 \tag{A.36}$$

and

$$s(u_1, p_2, m'_2) = 0.4 \cdot (2/3)(2/14)(0) + 0.6 \cdot (2/3)(4/14)(1/3) \approx 0.04 \tag{A.37}$$

$$a_{u_1, m'_2} = s(u_1, p_1, m'_2) + s(u_1, p_2, m'_2) \approx 0.08. \tag{A.38}$$

Since m'_1 has more weight than m'_2, thus, m'_1 (i.e., items with the same category or the same visual factor) is more important for "User 1" compared to m'_2 (i.e., items with the same brand or the same visual factor).

Example 7 (Item-MetaPath Association). Given the same HIN shown in Figure 1 and the same probabilistic meta-path $m'_1 = UP\{C \oplus V\}P$ with $\delta = 0.4$, Item-MetaPath association between p_1 and m'_1, a_{p_1,m'_1}, is computed as follows:

$$a_{p_1,m'_1} = \sum_{u \in \mathcal{U}_{p_1}} s(u, p_1, m'_1) = s(u_1, p_1, m'_1) + s(u_2, p_1, m'_1) \tag{A.39}$$

where

$$s(u_1, p_1, m'_1) = \delta \cdot g(u_1, m'_{1S})g(m'_{1S})g(m'_{1S}, p_1) + (1 - \delta) \cdot g(u_1, m'_{1V})g(m'_{1V})g(m'_{1V}, p_1) \tag{A.40}$$

and

$$s(u_2, p_1, m'_1) = \delta \cdot g(u_2, m'_{1S})g(m'_{1S})g(m'_{1S}, p_1) + (1 - \delta) \cdot g(u_2, m'_{1V})g(m'_{1V})g(m'_{1V}, p_1) \tag{A.41}$$

where

$$\begin{aligned}
g(u_1, m'_{1S}) &= \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_1) && \text{(A.42)}\\
&= Pr(p_1|u_1) + Pr(p_2|u_1) + Pr(p_3|u_1) + Pr(p_4|u_1) + Pr(p_5|u_1) && \text{(A.43)}\\
&= 1/3 + 1/3 + 0/3 + 0/3 + 0/3 = 2/3, && \text{(A.44)}\\
g(u_1, m'_{1V}) &= \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_1) && \text{(A.45)}\\
&= Pr(p_1|u_1) + Pr(p_2|u_1) + Pr(p_3|u_1) + Pr(p_4|u_1) + Pr(p_5|u_1) && \text{(A.46)}\\
&= 1/3 + 1/3 + 0/3 + 0/3 + 0/3 = 2/3, && \text{(A.47)}\\
g(u_2, m'_{1S}) &= \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_2) && \text{(A.48)}\\
&= Pr(p_1|u_2) + Pr(p_2|u_2) + Pr(p_3|u_2) + Pr(p_4|u_2) + Pr(p_5|u_2) && \text{(A.49)}\\
&= 0/4 + 0/4 + 1/4 + 1/4 + 0/4 = 1/2, && \text{(A.50)}\\
& && \text{(A.51)}
\end{aligned}$$

$$
\begin{aligned}
g(u_2, m'_{1V}) &= \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_2) && \text{(A.52)}\\
&= Pr(p_1|u_2) + Pr(p_2|u_2) + Pr(p_3|u_2) + Pr(p_4|u_2) + Pr(p_5|u_2) && \text{(A.53)}\\
&= 0/4 + 0/4 + 1/4 + 1/4 + 0/4 = 1/2, && \text{(A.54)}\\
g(m'_{1S}, p_1) &= \sum_{n_2 \in \mathcal{C}} Pr(p_1|n_2) = Pr(p_1|c_1) = 13 && \text{(A.55)}\\
g(m'_{1V}, p_1) &= \sum_{n_2 \in \mathcal{V}} Pr(p_1|n_2) = Pr(p_1|v_1) + Pr(p_1|v_2) = 1/3 + 0/3 = 1/3, && \text{(A.56)}\\
g(m'_{1S}) &= C(P, C) = 3/14 && \text{(A.57)}\\
g(m'_{1V}) &= C(P, V) = 4/14, && \text{(A.58)}
\end{aligned}
$$

Thus

$$s(u_1, p_1, m'_1) = 0.4 \cdot (2/3)(3/14)(1/3) + 0.6 \cdot (2/3)(4/14)(1/3) \approx 0.06 \quad \text{(A.59)}$$

and

$$s(u_2, p_1, m'_1) = 0.4 \cdot (1/2)(3/14)(1/3) + 0.6 \cdot (1/2)(4/14)(1/3) \approx 0.04 \quad \text{(A.60)}$$

$$a_{p_1, m'_1} = s(u_1, p_1, m'_1) + s(u_2, p_1, m'_1) \approx 0.1. \quad \text{(A.61)}$$

Similarly, for $m'_2 = UP\{B \oplus V\}P$, we can calculate

$$a_{p_1, m'_2} = \sum_{u \in \mathcal{U}_{p_1}} s(u, p_1, m'_2) = s(u_1, p_1, m'_2) + s(u_2, p_1, m'_2) \quad \text{(A.62)}$$

where

$$s(u_1, p_1, m'_2) = \delta \cdot g(u_1, m'_{2S}) g(m'_{2S}) g(m'_{2S}, p_1) + (1 - \delta) \cdot g(u_1, m'_{2V}) g(m'_{2V}) g(m'_{2V}, p_1) \quad \text{(A.63)}$$

and

$$s(u_2, p_1, m'_2) = \delta \cdot g(u_2, m'_{2S}) g(m'_{2S}) g(m'_{2S}, p_1) + (1 - \delta) \cdot g(u_2, m'_{2V}) g(m'_{2V}) g(m'_{2V}, p_1) \quad \text{(A.64)}$$

where

$$g(u_1, m'_{2S}) = \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_1) \tag{A.65}$$

$$= Pr(p_1|u_1) + Pr(p_2|u_1) + Pr(p_3|u_1) + Pr(p_4|u_1) + Pr(p_5|u_1) \tag{A.66}$$

$$= 1/3 + 1/3 + 0/3 + 0/3 + 0/3 = 2/3, \tag{A.67}$$

$$g(u_1, m'_{2V}) = \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_1) \tag{A.68}$$

$$= Pr(p_1|u_1) + Pr(p_2|u_1) + Pr(p_3|u_1) + Pr(p_4|u_1) + Pr(p_5|u_1) \tag{A.69}$$

$$= 1/3 + 1/3 + 0/3 + 0/3 + 0/3 = 2/3, \tag{A.70}$$

$$g(u_2, m'_{2S}) = \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_2) \tag{A.71}$$

$$= Pr(p_1|u_2) + Pr(p_2|u_2) + Pr(p_3|u_2) + Pr(p_4|u_2) + Pr(p_5|u_2) \tag{A.72}$$

$$= 0/4 + 0/4 + 1/4 + 1/4 + 0/4 = 1/2, \tag{A.73}$$

$$g(u_2, m'_{2V}) = \sum_{n_1 \in \mathcal{P}} Pr(n_1|u_2) \tag{A.74}$$

$$= Pr(p_1|u_2) + Pr(p_2|u_2) + Pr(p_3|u_2) + Pr(p_4|u_2) + Pr(p_5|u_2) \tag{A.75}$$

$$= 0/4 + 0/4 + 1/4 + 1/4 + 0/4 = 1/2, \tag{A.76}$$

$$\tag{A.77}$$

$$g(m'_{2S}, p_1) = \sum_{n_2 \in \mathcal{B}} Pr(p_1|n_2) = Pr(p_1|b_1) = 0, \tag{A.78}$$

$$g(m'_{2V}, p_1) = \sum_{n_2 \in \mathcal{V}} Pr(p_1|n_2) = Pr(p_1|v_1) + Pr(p_1|v_2) = 1/3 + 0/3 = 1/3, \tag{A.79}$$

$$g(m'_{2S}) = C(P, B) = 2/14, \tag{A.80}$$

$$g(m'_{2V}) = C(P, V) = 4/14. \tag{A.81}$$

Thus,

$$s(u_1, p_1, m'_2) = 0.4 \cdot (2/3)(2/14)(0) + 0.6 \cdot (2/3)(4/14)(1/3) \approx 0.04 \tag{A.82}$$

and

$$s(u_2, p_1, m'_2) = 0.4 \cdot (1/2)(2/14)(0) + 0.6 \cdot (1/2)(4/14)(1/3) \approx 0.03 \qquad \text{(A.83)}$$

$$a_{p_1, m'_2} = s(u_1, p_1, m'_2) + s(u_2, p_1, m'_2) \approx 0.07. \qquad \text{(A.84)}$$

Since m'_1 has more weight than m'_2, thus, m'_1 (i.e., items with the same category or the same visual factor) is more important for "T-shirt A" compared to m'_2 (i.e., items with the same brand or the same visual factor).