Information Processing and Management 59 (2022) 102787

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Personalized Recommendation via Multi-dimensional Meta-paths Temporal Graph Probabilistic Spreading

Yang Wang [a,b], Lixin Han [a,*], Quiping Qian [c], Jianhua Xia [a], Jingxian Li [d]

[a] College of Computer and Information, Hohai University, Nanjing 211100, PR China
[b] School of Computer and Information, Anqing Normal University, Anqing 246133, PR China
[c] School of Foreign Languages, Anqing Normal University, Anqing 246133, PR China
[d] School of Software Engineering, Jinling Institute of Technology, Nanjing 211169, PR China

ARTICLE INFO

Keywords:
Personalized recommendation
Meta-path
Probabilistic spreading
Temporal graph
Boosting strategy

ABSTRACT

Since meta-paths have the innate ability to capture rich structure and semantic information, meta-path-based recommendations have gained tremendous attention in recent years. However, how to composite these multi-dimensional meta-paths? How to characterize their dynamic characteristics? How to automatically learn their priority and importance to capture users' diverse and personalized preferences at the user-level granularity? These issues are pivotal yet challenging for improving both the performance and the interpretability of recommendations. To address these challenges, we propose a personalized recommendation method via Multi-Dimensional Meta-Paths Temporal Graph Probabilistic Spreading (MD-MP-TGPS). Specifically, we first construct temporal multi-dimensional graphs with full consideration of the interest drift of users, obsolescence and popularity of items, and dynamic update of interaction behavior data. Then we propose a dimension-free temporal graph probabilistic spreading framework via multi-dimensional meta-paths. Moreover, to automatically learn the priority and importance of these multi-dimensional meta-paths at the user-level granularity, we propose two boosting strategies for personalized recommendation. Finally, we conduct comprehensive experiments on two real-world datasets and the experimental results show that the proposed MD-MP-TGPS method outperforms the compared state-of-the-art methods in such performance indicators as precision, recall, F1-score, hamming distance, intra-list diversity and popularity in terms of accuracy, diversity, and novelty.

1. Introduction

Recommender systems are one of the most promising and powerful solutions to the problems of information overload and formation disorientation. However, as mainstream and dominant algorithms, the collaborative filtering (CF) and bipartite graph-based recommendation suffer from the vexing problems of cold start and data sparsity. With the surge of Web 2.0 applications, various kinds of side information has been leveraged to improve the recommendation quality, resulting in significant attention to heterogeneous graphs/networks (Belém, Heringer, Almeida, & Gonçalves, 2019; Ji, Shi, Fang, Kong, & Yin, 2020; Li, Wang, Lyu, & Shi, 2020; Sánchez & Bellogín, 2019; Xin & Wu, 2020). An illustrative heterogeneous graph is demonstrated in Fig. 1, which consists of a large

* Corresponding author.
E-mail address: lixinhan2002@aliyun.com (L. Han).

https://doi.org/10.1016/j.ipm.2021.102787
Received 24 April 2021; Received in revised form 5 October 2021; Accepted 10 October 2021
Available online 18 October 2021

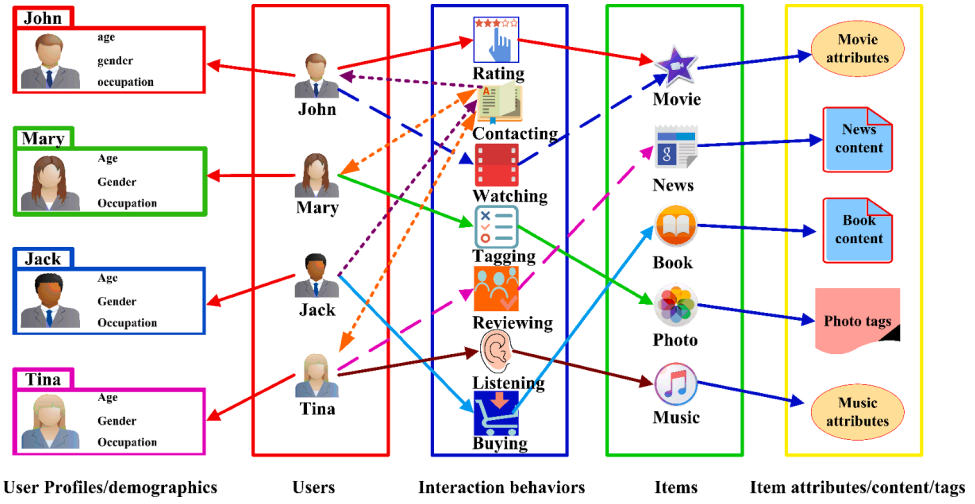


Fig. 1. An illustrative heterogeneous graph example.

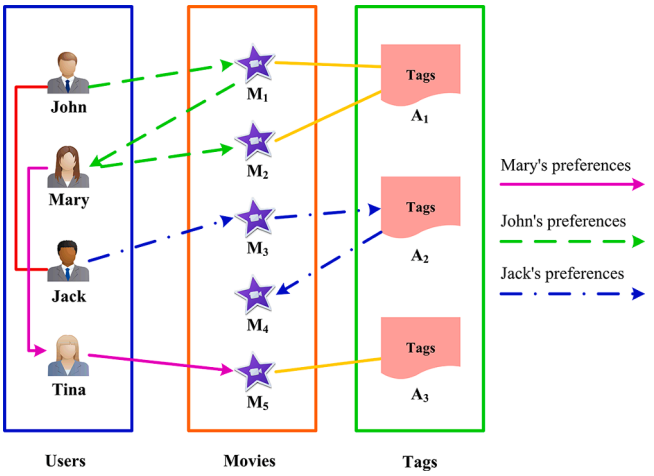


Fig. 2. An example reflecting diverse users' preferences.

collection of various kinds of nodes, including users, movies, news, books, photos, and music as well as diverse interaction behaviors (e.g. users rate movies, contact friends, watch movies, tag photos, review news, listen to music, buy books etc.). In the graph there is also a great deal of side information like users' profile/demographics, books' content, movies' attributes, photos' tags, and social trust relationship etc. Such rich structural and semantic information brings an attractive yet challenging task. That is how to make full use of it to make desirable and inspiring recommendations.

Recent years have witnessed tremendous attention attached to heterogeneous graph-based recommendation works (Sun, Han, Yan, Yu, & Wu, 2011; Shi, Kong, Huang, Yu, & Wu, 2014; Shi et al. 2015; Hu, Shi, Zhao, & Yu, 2018; Shi, Hu, Zhao, & Yu, 2019; Zhang & Chen, 2020; Mitrović & De Weerdt, 2020). These fine-grained methods have made great progress possible in improving the performance of recommendation. However, the research on meta-path-based recommendations is still far from enough in either leveraging multi-dimensional meta-paths to identify the user-level preferences or capturing the dynamic characteristics of multi-dimensional meta-paths.

(1) There is still a lack of research on using multi-dimensional meta-paths to capture users' miscellaneous and personalized preferences. The incipient work mainly pays attention to evaluating the similarity/relevance of heterogeneous nodes under different meta-paths and helping to select neighbor nodes for the collaborative filtering algorithm. However, capturing structural characteristics of meta-paths only, without taking full advantage of the rich semantics represented by them results in reduced performance and poor interpretability of recommendations. In recent years, most research efforts focus on utilizing the feature representation/network embedding methods (Wang et al., 2019; Xie et al., 2021) to map heterogeneous nodes' features into a unified low-dimensional space in the form of feature vectors, which are then aggregated for joint matrix decomposition (Chen, Zheng, Liu, & Yu, 2018; Jiang, Ding, & Fu, 2020) or fed to neural networks/deep learning networks with attention mechanisms (Molaei, Zare, & Veisi, 2020; Shi et al., 2021)

to learn the importance of multiple meta-paths. Different from the above methods, based on the fact that each user has diverse and personalized preferences in accord with preferring to some special meta-paths, we concentrate on using multi-dimensional meta-paths to capture users' personalized preferences at the user-level granularity and expect to automatically learn the priority and importance of these meta-paths for each user.

Consider a movie scenario as shown in Fig. 2:

(i) Mary favors those movies that her friends recommend. For instance, suppose Tina is Mary's friend and Tina has watched movie M_5, it will be of high possibility for Mary's preferring to select M_5, following the path Mary→Tina→M_5 (marked with the pink solid line). In this case the corresponding meta-path is User→User→Movie.
(ii) John favors those movies viewed by users with similar preferences. Suppose John and Mary have both watched movie M_1 and Mary has also watched Movie M_2, it is likely for John to select M_2, following the path John→M_1→Mary→M_2 (marked with the green dotted line). Under this circumstance, the corresponding meta-path is User→Movie→User→Movie.
(iii) Jack favors those movies tagged the same tag as those he has watched. Suppose Jack has watched movie M_3 and both M_3 and M_4 are tagged as A_2, Jack will probably prefer M4, following the path Jack→M_3→A_2→M_4 (marked with the blue dot dash line), corresponding to the meta-path User→Movie→Tag→Movie.

As exampled in the above-mentioned users' behaviors, it can be concluded that each meta-path represents a certain special semantic meaning and meta-paths have inherent capability to capture the rich semantic meanings. Mary, John, and Jack all have their own personalized preferences which are represented by some particular meta-paths. In that case, how to composite these multi-dimensional meta-paths? Is there a means to automatically learn the priority and the importance of these meta-paths to capture users' personalized preferences at the user-level granularity? These are crucial yet challenging tasks for improving not only the performance but also the interpretability of the recommendation.

(2) Most of the existing heterogeneous graph-based recommendation works regard graphs as static, utilizing such approaches as meta-graph theory (Xie, Zheng, Chen, & Zheng, 2021; Zhao, Yao, Li, Song, & Lee, 2017), networking embedding methods (Dong, Chawla, & Swami, 2017; Mitrović & De Weerdt, 2020; Fu, Lee, & Lei, 2017; Zhang, Huang, Tan, Sun, & Zhou, 2021), and neural networks/deep learning architectures (Molaei, Zare, & Veisi, 2020; Shi et al., 2021) to mine networks structure. However, they neglect the dynamic nature (time dependence and volatility) of heterogeneous graphs. As time context information has been proved the most crucial ingredient in the improvement of recommendation quality, we argue that heterogeneous graphs are dynamic networks with a continual evolution with time because of the interest drift of users, obsolescence and popularity of items, and dynamic update of interaction behavior data.

To fill up the gap, in this paper, our goal is to (i) incorporate rich semantic meanings associated with multi-dimensional meta-paths to improve recommendation performance and (ii) automatically learn the priority and importance of these meta-paths to capture users' personalized preferences in the user-level granularity and (iii) consider the dynamic nature of multi-dimensional meta-paths. Specifically,

(1) With full consideration of the interest drift of users, obsolescence and popularity of items, and dynamic update of interaction behavior data, we characterize the dynamic characteristics of heterogeneous graphs and construct temporal multi-dimensional graphs.
(2) We incorporate rich semantics associated with multi-dimensional meta-paths and propose a dimension-free temporal graph probabilistic spreading framework. Taking items recommendation in a tripartite graph as an example, we elaborate 2-dimensional/3-dimensional meta-path-based temporal graph probabilistic spreading process.
(3) We propose two personalized recommendation strategies to automatically learn the priority and importance of these meta-paths in the user-level granularity. One applies the boosting strategy of a vote and accumulation mechanism and the other applies the boosting strategy of a maximum and rearrangement mechanism.

In summary, our main contributions are:

(1) In terms of methodology, we propose a personalized recommendation method via multi-dimensional meta-paths temporal graph probabilistic spreading.
(2) From a theoretical point of view, we construct temporal multi-dimensional graphs with the consideration of the interest drift of users, obsolescence and popularity of items, and dynamic update of interaction behavior data.
(3) From a technological perspective, we propose a dimension-free temporal graph probabilistic spreading framework via multi-dimensional meta-paths and two personalized recommendation strategies.
(4) From a practical viewpoint, we propose a MD-MP-TGPS method that is competitive against other state-of-the-art baselines in the following performance indicators, namely precision, recall, F1-score, hamming distance, intra-list diversity, and popularity in terms of accuracy, diversity, and novelty.

The rest of this paper is organized as follows. We review the related work in Section 2. Section 3 provides some notations as well as background on meta-paths and graph probabilistic spreading. Section 4 elaborates the proposed MD-MP-TGPS method. Section 5 describes the experimental process and demonstrates experimental results. Section 6 summarizes the findings and discusses future work.

Table 1
Notations used in the paper. The left column represents the abbreviation and in the right column the description of the abbreviated term.

Notation	Description
$\mathscr{G}$	heterogeneous graph
$\mathscr{N}$	node set
$\mathscr{E}$	edge set
S	graph schema
$\mathscr{C}$	node type set
$\mathscr{R}$	edge type set
$\mathscr{P}$	meta-path
U	user set
I	item set
A	tag set
t	time of interaction behavior
$T/\Delta t$	long-time window/short-time widow
$a_{\alpha i}/a_{io}/a_{\alpha o}$	edge weight between U_i and I_α/U_i and A_o/I_α and A_o in time-unaware graph
$r_{\alpha i}/r_{io}/r_{\alpha o}$	edge weight between U_i and I_α/U_i and A_o/I_α and A_o in temporal graph
τ/φ	decay rate/bias offset
η	long time-window parameter
f	resource score
K	node degree
L	length of recommendation list
$\mathscr{L}^{U_i}$	final recommendation list for U_i
$\mathscr{L}_p^{U_i}$	recommendation list based on meta-path p for U_i
$\mathscr{L}_{\mathscr{P}}^{U_i}$	recommendation list set based on multi-dimensional meta-paths for U_i

2. Related work

Since our focus is on meta-path-based personalized recommendation via temporal graph probabilistic spreading, we mainly discuss the related work from three aspects: graph probabilistic spreading, time-aware recommendation, and meta-path-based recommendation.

2.1. Graph probabilistic spreading

The probabilistic spreading model (ProbS) was initially proposed by Zhou et al. (2010) to solve the diversity-accuracy dilemma in bipartite graph-based recommendation by combining another spreading model HeatS (heat spreading). Huang, Chen, Wang, and Mei (2014) built a user-query-video tripartite graph and leveraged the graph propagation for personalized video recommendation. Poulain and Tarissan (2020) composited user-product and product-category bipartite graphs for a formation of the user-product-category tripartite graph and then adopted a random walk algorithm to traverse through it. Li, Tang, and Chen (2017) fused the regularized matrix factorization with a user–item–tag weighted diffusion model to improve recommendation performance. Johnson and Ng (2017) used tripartite graphs and Markov process to enhance the ability of recommending long-tail items. Huang, Shen and Meng (2019) used the influence diffusion mechanism to recommend items dynamically. Zhou and Han (2019) modified resource allocation process in a user-preference-item tripartite graph for the purpose of matching the target user. These methods mainly apply such spreading algorithms as random walk, Markov process, resource allocation, and mass diffusion in bipartite/tripartite graphs to improve recommendation quality. However, the meta-path adopted in these graph diffusion algorithms is relatively simple, resulting in the neglect of rich semantics of multi-dimensional meta-paths. Moreover, most of the above algorithms also ignore the dynamic characteristics of heterogeneous graphs with evolution with time.

2.2. Time-aware recommendation

As a significant factor, temporal dynamics has been modeled to help Bell & Koren group win the contest on the explicit feedback dataset Netflix (Koren, 2010; Koren & Bell, 2015). To characterize long-term and short-term preferences of users on implicit feedback datasets, Xiang et al. (2010) claimed a session-based temporal recommendation method. Based on the user-based collaborative filtering, Campos, Díez, and Cantador (2014) proposed a time decay function for temporal recommendation. Najafabadi, Mohamed and Onn (2019) considered the time impact and modeled user profiles using the graph-based model. Hu, Han, Lin, Huang, and Zhang (2019) used an attention-driven CNN to acquire learners' interactive behaviors in a tripartite graph for peer recommendation. Sánchez and Bellogín (2020) proposed a soften time decay function to measure the similarity between two users for recommendation. Considering users' temporal behavior, a recurrent Poisson factorization framework was presented by expanding the Poisson factorization with a Poisson process (Hosseini et al., 2020). Wang and Han (2020) suggested a time-dependent network-based inference

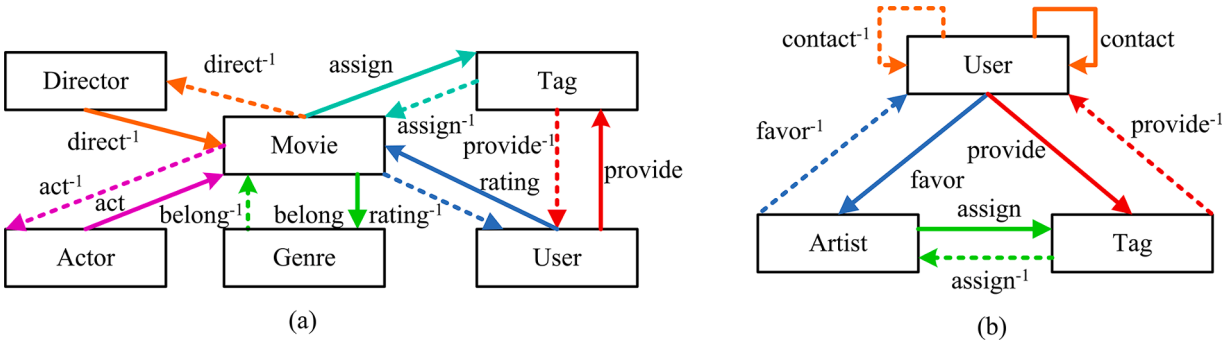


Fig. 3. The graph schemas of MovieLens and Last.fm datasets.

recommendation method in bipartite networks. Different from the above-mentioned methods whose focus is mainly on modeling user preferences, we not only consider interest drift of users, but also take obsolescence and popularity of items, and dynamic update of interaction behavior data into account.

2.3. Meta-path-based recommendation

Sun, Han, Yan, Yu, and Wu (2011) first introduced the concept of the meta-path and proposed Pathsim to measure the similarity /relevance between homogeneous nodes. To solve the problem of relevance search between heterogeneous node pairs, Shi, Kong, Huang, Yu, & Wu, 2014 put forward a novel algorithm HeteSim based on the constrained meta-path. Shi et al. (2015) proposed a personalized recommendation method SemRec based on meta-path to predict the users' rating scores on items. A network embedding based method HERec was proposed to make use of side information for recommendation in heterogeneous information networks (Shi, Hu, Zhao, & Yu (2019). Different from SemRec and HERec focusing on rating prediction, Hu, Shi, Zhao, and Yu (2018) leveraged the meta-path context to set up a deep neural co-attention model for top-N recommendation. Zhang and Chen (2020) proposed a network embedding framework based on multiple meta-paths for heterogeneous networks. Most of these methods regard heterogeneous networks as static graphs and neglect the dynamic nature with evolution with time. We comprehensively consider interest drift of users, obsolescence and popularity of items, and dynamic update of interaction behavior data. Recently, Wang et al. (2020) proposed a dynamic network embedding learning model DyHNE with proximities based on meta-paths. Different from these studies focusing on adopting network embedding/represent learning methods, and neural networks/deep learning architectures to tune or learn hyper-parameters, we propose two personalized recommendation strategies to automatically learn the priority and importance of these meta-paths at the user-level granularity.

3. Preliminaries

In this section, we first summarize the notations used throughout the paper in Table 1 and then briefly introduce the background on meta-paths and graph probabilistic spreading. The detailed description of them can be respectively found in Sun, Han, Yan, Yu, and Wu (2011) and Zhou et al. (2010).

3.1. Basic definitions

Definition 1. (Heterogeneous graph). A graph is generally denoted as $\mathcal{G}(\mathcal{N},\mathcal{E})$, with $\mathcal{N}$ representing the node set and $\mathcal{E}$ the edge set. Two functions are used to illustrate the mapping relation between graph $\mathcal{G}$ and its corresponding schema $S(\mathscr{C},\mathscr{R})$. $\mathrm{F_n}:\mathcal{N}\rightarrow\mathscr{C}$ describes to which node type each node in node set $\mathcal{N}$ belongs. Similarly, $\mathrm{F_e}:\mathcal{E}\rightarrow\mathscr{R}$ ascertains to which edge type each edge in edge set $\mathcal{E}$ belongs. If $|\mathscr{C}|+|\mathscr{R}|>2$, graph $\mathcal{G}$ is a heterogeneous graph, otherwise it is a homogeneous one. Since heterogeneous graphs represent a variety of rich structure characteristics, relationships, and semantics, we are to study them in this paper rather than homogeneous graphs.

Definition 2. (Graph schema). Denoted as $S(\mathscr{C},\mathscr{R})$, a graph schema is the meta template of a particular graph $\mathcal{G}$. It specifies the node types and edge types (i.e. relationships) contained in the graph. For instance, the graph schemas of two real-world datasets MovieLens and Last.fm are shown in Fig. 3. Taking Last.fm schema as an example, $\mathscr{C}$ includes three node types: {User, Artist, Tag} and $\mathscr{R}$ contains eight relationships among these node types (e.g. users contact their own friends, users favor artists, artists are assigned tags, etc).

Definition 3. (Meta-path). A meta-path $\mathscr{P}$ is conceptualized on the basis of the graph schema $S(\mathscr{C},\mathscr{R})$. Generally, it represents the traversing process from the start node type to the target node type in the form of an ordinal sequence $\mathscr{C}_1\xrightarrow{\mathscr{R}_1}\mathscr{C}_2\xrightarrow{\mathscr{R}_2}\cdots\xrightarrow{\mathscr{R}_l}\mathscr{C}_{l+1}$. A composite relationship $\mathscr{R}_{1,2,\cdots l}=\mathscr{R}_1\circ\mathscr{R}_2\circ\cdots\circ\mathscr{R}_l$ suggests the semantic meaning of the meta-path where $\circ$ is a composite operator. For example,

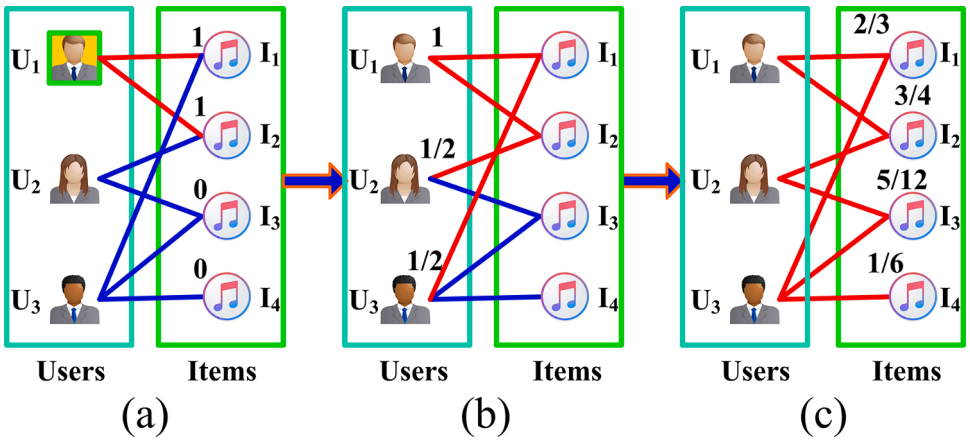


Fig. 4. The process of a bipartite graph-based probabilistic spreading.

as shown in Fig. 3(b), the meta-paths User $\xrightarrow{\text{favor}}$ Artist $\overset{\text{favor}^{-1}}{\rightarrow}$ User and User $\xrightarrow{\text{favor}}$ Artist $\xrightarrow{\text{assign}}$ Tag $\overset{\text{assign}^{-1}}{\rightarrow}$ Artist $\overset{\text{favor}^{-1}}{\rightarrow}$ User respectively indicate that both of the two users favor the same artist and two users favor the artists who are tagged the same.

Definition 4. (Multi-dimensional meta-path)**.** If the number of different node types in a meta-path is N, then this meta-path is intituled as an N-dimensional meta-path. For example, as shown in Fig. 3(b), User → User → Useris a 1-dimensional meta-path, User → Artist → Useris a 2-dimensional meta-path, both Artist → Tag → User → Artistand Artist → User → Tag → Artistare 3-dimensional meta-paths, and so on and so forth.

3.2. Bipartite graph-based probabilistic spreading

The bipartite graph-based probabilistic spreading model (ProbS), also known as network-based inference (NBI), utilizes a random walker to traverse on user-item bipartite graphs. From the perspective of physics, it is a two-step mass diffusion model, obeying the law of mass conservation. ProbS uses the concept of resource to represent a target user' preferences and utilizes the physical process of mass diffusion to spread preferences in the graph to obtain his/her preferences degree for each item, which has good motivation, solid physical meaning and theoretical availability. Moreover, it can effectively alleviate the problems of data sparsity and scalability in traditional collaborative filtering, achieving better recommendation accuracy at lower computational cost compared with the collaborative filtering and global ranking method. In addition, it is independent of the feature vectors of items, which overcomes the difficulty of content-based recommendation in the feature extraction of video and audio items (Campana & Delmastro, 2017; Lü et al., 2012; Yu, Zeng, Gillard, & Medo, 2016).

Considering an undirected bipartite graph $\mathscr{G}(\mathscr{N},\mathscr{E}), \mathscr{N} = \{v|v \in U \vee v \in I \wedge U \cap I = \varnothing\}$consists of two disjoint sets: user set U and item set I. Given $U = \{U_1, U_2, ..., U_m\}, I=\{I_1, I_2, ..., I_n\}$, $\mathscr{E}$ is represented as $\{e|e = (U_i, I_\alpha, a_{\alpha i}) \wedge U_i \in U \wedge I_\alpha \in I\}$ where $a_{\alpha i}$ is the weight of the edge between U_i and I_α.

In this section, we briefly review the spreading and recommendation process of ProbS. Given a target user U_i, Probs first gives an initial preference value representing U_i's preferences (usually called initial resource) to each item. If the items are selected by U_i, the initial resource is set as scalar value 1, otherwise 0. In this way, each item gets a certain amount of initial resources, denoted as $f(I)$. U_i's preferences are then spread across the user-item bipartite graph in the form of resources. And then each item equally transfers the obtained resource it possessed to neighboring users. As a result, the resource of U_i received from all neighboring items is:

$$f^{'}(U_i) = \sum_{\beta=1}^{n} \frac{a_{\beta i} f(I_\beta)}{K(I_\beta)}, \tag{1}$$

where $f(I_\beta)$ is the initial resource of I_β, $K(I_\beta) = \sum_{i=1}^{m} a_{\beta i}$ is the degree of I_β.

After that, each user flows the owned resource back to his/her collected items. The final resource of I_α obtained from the two-step probabilistic spreading based on $\mathscr{G}$ is:

$$f''(I_\alpha) = \sum_{i=1}^{m} \frac{a_{\alpha i} f(U_i)}{K(U_i)} = \sum_{i=1}^{m} \frac{a_{\alpha i}}{K(U_i)} \sum_{\beta=1}^{n} \frac{a_{\beta i} f(I_\beta)}{K(I_\beta)} = \sum_{\beta=1}^{n} \frac{1}{K(I_\beta)} \sum_{i=1}^{m} \frac{a_{\alpha i} a_{\beta i}}{K(U_i)} f(I_\beta), \tag{2}$$

where $K(U_i) = \sum_{\beta=1}^{n} a_{\beta i}$ is the degree of U_i.

After the above two-step process of resource spreading, each item has a final resource, which represents the item's recommendation score given by the graph probabilistic spreading algorithm for the target user. The higher the score an item gets is, the more interest the target user has in it (Campana & Delmastro, 2017). Finally, all U_i's unvisited items are sorted in the descending order of $f''(I_\alpha)$, and the

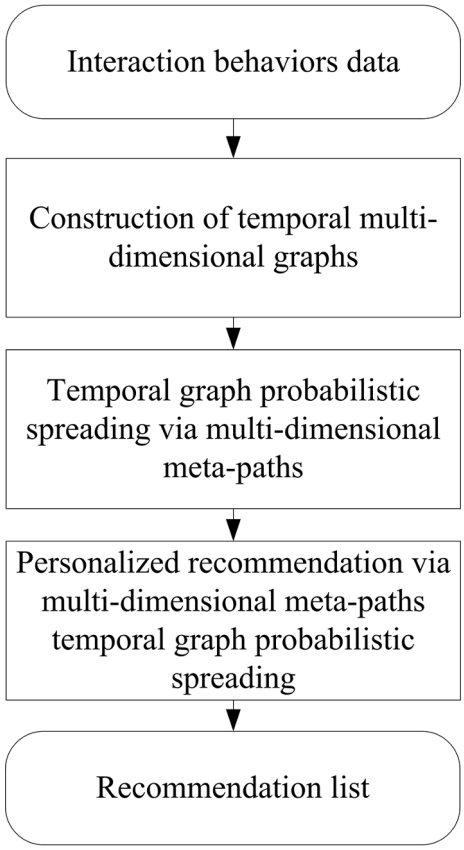


Fig. 5. The framed illustration of MD-MP-TGPS.

top L items are to be recommended to U_i as a recommendation list.

To concretely reveal the above-mention resource allocation process, we make an implicit feedback-based bipartite graph $\mathscr{G}$ (comprised of three users and four items) as an example, as shown in Fig. 4.

In graph $\mathscr{G}$, if U_i has collected I_α, $a_{\alpha i} = 1$, otherwise $a_{\alpha i} = 0$ (In an explicit feedback-based bipartite graph, if U_i has collected I_α, $a_{\alpha i}$ equals the U_i's rating score on I_α, otherwise 0). Supposing U_1 is the target user, we initially assign a unit resource to all items collected by U_1 (i.e. I_1 and I_2), as shown in Fig. 4(a). Therefore the initial resource vector of items is:

$$\overrightarrow{\mathbf{f}} = (1, 1, 0, 0) \tag{3}$$

Then, as shown in Fig. 4(b), I_1 and I_2 evenly distribute resources to their own neighboring users, resulting in the medi-distribution resource vector of users is:

$$\overrightarrow{\mathbf{f}'} = (1, 1/2, 1/2) \tag{4}$$

As shown in Fig. 4(c) that followed, the resources of U_1, U_2, U_3 re-distribute back to items. The final resource vector of items is:

$$\overrightarrow{\mathbf{f}''} = (2/3, 3/4, 5/12, 1/6) \tag{5}$$

Finally, I_3 is recommended to the target user U_1 because it has a higher final resource score (here is 5/12) than that of I_4 (here is 1/6).

4. The proposed MD-MP-TGPS method

In this paper, we aim at: (i) fully considering the dynamic nature of heterogeneous graphs and (ii) incorporating rich semantic meaning associated with multi-dimensional meta-paths to improve recommendation performance and (iii) automatically learning the priority and importance of these meta-paths to capture users' personalized preferences at the user-level granularity. To this end, we propose a personalized recommendation method via Multi-Dimensional Meta-Paths Temporal Graph Probabilistic Spreading (MD-MP-TGPS). To be specific, we first construct temporal multi-dimensional graphs with full consideration of the interest drift of users, obsolescence and popularity of items, and dynamic update of interaction behavior data (Section 4.1). Then we propose a temporal

Table 2
The mapping of a segment of three-way interaction data among users, items, and tags. (a) represents the original behavior data, and (b) refers to the mapped result of the data in (a) after the time window processing calculated by Eq (6).

(a) original behavior data				(b) the mapped data			
U	I	A	t	U	I	A	T
U_1	I_1	A_1	t_1	U_1	I_1	A_1	T_0
U_1	I_1	A_2	t_2	U_1	I_1	A_2	T_2
U_1	I_3	A_1	t_3	U_1	I_3	A_1	T_3
U_2	I_2	A_2	t_4	U_2	I_2	A_2	T_0
U_2	I_2	A_3	t_5	U_2	I_2	A_3	T_1
U_2	I_3	A_3	t_6	U_2	I_3	A_3	T_2

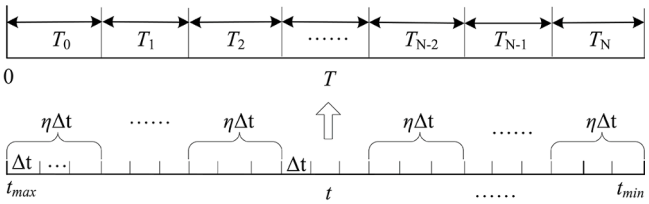


Fig. 6. The partition of time windows. Δt refers to a short-time window, and $T{=}\eta\Delta t$ denotes a long-time window.

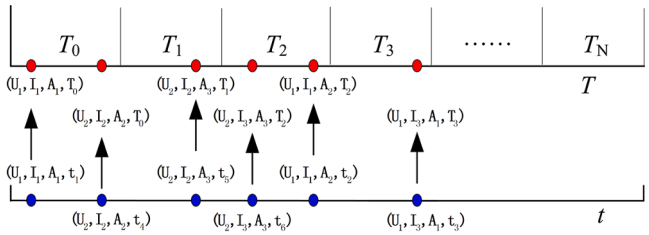


Fig. 7. The schematic diagram for mapping each behavior data $(U_i, I_\alpha, A_o, t_x)$ (blue points) onto $(U_i, I_\alpha, A_o, T_y)$ (red points) (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article).

graph probabilistic spreading framework via multi-dimensional meta-paths (Section 4.2). At last, we propose two boosting strategies for personalized recommendation to automatically learn the priority and importance of these multi-dimensional meta-paths at the user-level granularity (Section 4.3). The framed illustration of MD-MP-TGPS is as shown in Fig. 5.

4.1. Construction of temporal multi-dimensional graphs

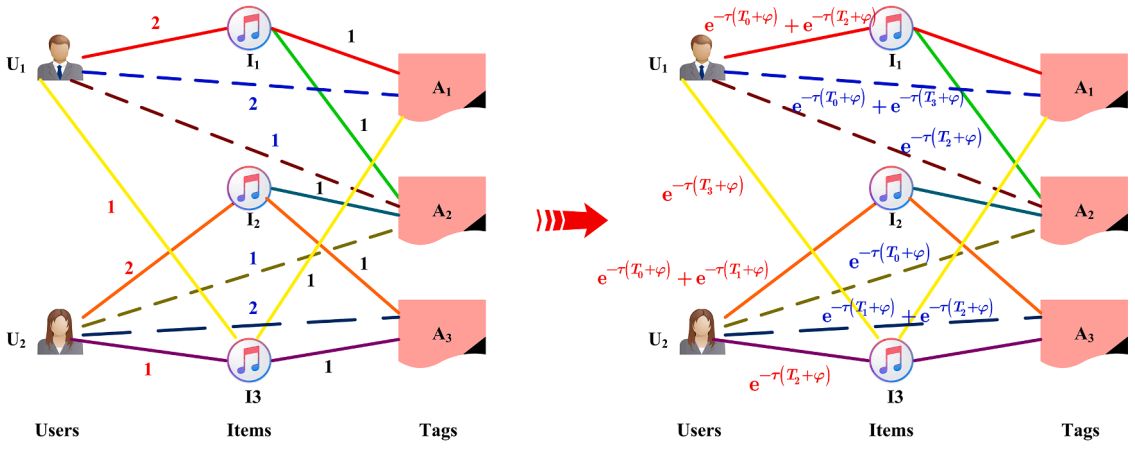
The existing heterogeneous graph-based recommendation works mainly focus on utilizing the graph theory, network embedding method, and neural networks/deep learning architectures to mine structure characteristics of networks for similarity measurement/relevance search, link prediction, classification, clustering, community detection, and recommendation, which are still inadequate for modeling dynamic nature of heterogeneous graphs.

As time context information has been proved as the most crucial ingredient for the improvement of recommendation quality, we argue that heterogeneous graphs are dynamic with evolution with time. In this paper, following the work (Wang & Han, 2020), we not only consider the interest drift of users, but also take obsolescence and popularity of items, and dynamic update of interaction behavior data into account.

For the sake of description, we take the construction of a temporal 3-dimensional graph as an example. In a segment of three-way interaction data among two users, three items, and three tags as shown in Table 2(a), $U{=}\{U_1, U_2, \ldots, U_m\}$ is the user set, $I{=}\{I_1, I_2, \ldots, I_n\}$ is the item set, $A{=}\{A_1, A_2, \ldots, A_k\}$ is the tag set, and t represents the time when the interaction behavior has turned up. Thus the behavior data can be represented by a series of quadruples (U, I, A, t).

Based on the work (Wang & Han, 2020), as shown in Fig. 6, the closest and earliest time pointing to the present is respectively represented as $t_{\max}$, $t_{\min}$. Then the time interval $[t_{\min}, t_{\max}]$ is partitioned into small intervals named as short-time window at a fixed or variable time length, denoted as Δt (here we set $\Delta t{=}1$ day$=$86400 seconds). A long-time window T composes of η short-time windows (i.e. $T{=}\eta\Delta t$).

And then as shown in Fig. 7, each behavior data with interaction time t_x is mapped onto its corresponding long-time window T_y by a mapping function f: $(U_i, I_\alpha, A_o, t_x) \rightarrow (U_i, I_\alpha, A_o, T_y)$, where T_y is defined as follows:



(a) $\mathcal{G}_{\mathrm{u}}$: time-unaware 3-dimensional graph **(b) $\mathcal{G}_{\mathrm{t}}$: temporal 3-dimensional graph**

Fig. 8. Construction of a temporal graph. In a temporal multi-dimensional graph, the intensity of an interaction behavior between two nodes is decayed with a nonlinear exponential decay function. The weight of an edge accumulates the intensities on all direct paths between two nodes.

$$T_y = \mathrm{int}\left(\frac{t_{\max} - t_x}{86400 \times \eta}\right). \tag{6}$$

The result of the original behavior data in Table 2(a) after the mapping process shown in Fig. 7 is shown in Table 2(b).

As to three-way interaction data shown in Table 2(a), if the time effect is not taken into account, a time-unaware 3-dimensional graph, denoted as $\mathcal{G}_{\mathrm{u}}(U,I,A,t)$, is constructed as shown in Fig. 8(a). Generally, the weight of an edge implies the intensity of relationship between two nodes. In time-unaware graph $\mathcal{G}_{\mathrm{u}}$, the weight is represented by the number of direct paths between two nodes. For example, U_1 has used the A_1 and A_2 to tag I_1 at time t_1 and t_2. It indicates there are two direct paths connecting U_1 and I_1 with intensity being $a_{U_1I_1}{}^{t_1} = 1$ and $a_{U_1I_1}{}^{t_2} = 1$ respectively. As a result, weight $a_{U_1I_1}$ of edge U_1-I_1 equals 2. This weighting way ignores not only time dependence but also volatility of heterogeneous graphs, which downgrades the recommendation performance to some degree. To fully reflect time-varying nature of users' interaction behaviors and actually capture the semantic meaning of various meta-paths, we take such factors into account as interest drift of users, obsolescence and popularity of items, and dynamic update of interaction behavior data to construct a temporal 3-dimensional graph, denoted as $\mathcal{G}_{\mathrm{t}}(U,I,A,T)$, shown in Fig. 8(b).

In temporal 3-dimensional graph $\mathcal{G}_{\mathrm{t}}$, the intensity of an interaction behavior between U_i and I_α within long-time window T, denoted as $r_{\alpha i}{}^{T}$, is represented as follows:

$$r_{\alpha i}{}^{T} = a_{\alpha i}{}^{t} e^{-\tau(T+\varphi)}, \tag{7}$$

where τ indicates the decay rate of a nonlinear exponential time decay function (Wang & Han, 2020) and φ is the bias offset.

Analogously, we can obtain $r_{io}{}^{T}$ (the intensity between U_i and A_o within long-time window T) and $r_{\alpha o}{}^{T}$ (the intensity between I_α and A_o within long-time window T) noted as:

$$\begin{aligned} r_{io}{}^{T} &= a_{io}{}^{t} e^{-\tau(T+\varphi)}, \\ r_{\alpha o}{}^{T} &= a_{\alpha o}{}^{t} e^{-\tau(T+\varphi)}. \end{aligned} \tag{8}$$

As the intensity of each direct path in time-unaware graph $\mathcal{G}_{\mathrm{u}}$ equals 1, (i.e. $a_{\alpha i}{}^{t} = 1,\ \ a_{io}{}^{t} = 1, a_{\alpha o}{}^{t} = 1$), then $r_{\alpha i}{}^{T} = e^{-\tau(T+\varphi)}$, $r_{io}{}^{T} = e^{-\tau(T+\varphi)}$, $r_{\alpha o}{}^{T} = e^{-\tau(T+\varphi)}$ in temporal graph $\mathcal{G}_{\mathrm{t}}$. Finally, the weight of an edge accumulates the intensities on all direct paths between two nodes. The weight of the edge between user U_i and I_α, denoted as $r_{\alpha i}$, is written as:

$$r_{\alpha i} = \sum_{P_{\alpha i}} r_{\alpha i}{}^{T}, \tag{9}$$

where $P_{\alpha i}$ is the set of all direct paths between U_i and I_α.

Analogously, we can obtain r_{io} (the edge weight between U_i and A_o) and $r_{\alpha o}$ (the edge weight between I_α and A_o) as:

$$\begin{aligned} r_{io} &= \sum_{P_{io}} r_{io}{}^{T}, \\ r_{\alpha o} &= \sum_{P_{\alpha o}} r_{\alpha o}{}^{T}. \end{aligned} \tag{10}$$

Weight $r_{U_1I_1}$ of edge U_1-I_1 is equal to $\mathrm{e}^{-\tau(T_0+\varphi)} + \mathrm{e}^{-\tau(T_2+\varphi)}$ as shown in Fig. 8(b) because there are two paths connecting U_1 and I_1 and their intensities are respectively $r_{U_1I_1}{}^{T_0} = \mathrm{e}^{-\tau(T_0+\varphi)}$ and $r_{U_1I_1}{}^{T_2} = \mathrm{e}^{-\tau(T_2+\varphi)}$. To keep neat, we only give the edge weights between users and items, along with those between users and tags in Fig. 8(b). The detailed comparison of the edge weights in time-unaware graph $\mathcal{G}_{\mathrm{u}}$ and

Table 3
The comparison of the weights of edges between users and items, users and tags, items and tags respectively in time-unaware graph $\mathcal{G}_u$ and temporal graph $\mathcal{G}_t$.

edge	weights in $\mathcal{G}_u$	weights in $\mathcal{G}_t$	edge	weights in $\mathcal{G}_u$	weights in $\mathcal{G}_t$	edge	weights in $\mathcal{G}_u$	weights in $\mathcal{G}_t$
U_1-I_1	2	$e^{-\tau(T_0+\varphi)} + e^{-\tau(T_2+\varphi)}$	U_1-A_1	2	$e^{-\tau(T_0+\varphi)} + e^{-\tau(T_3+\varphi)}$	I_1-A_1	1	$e^{-\tau(T_0+\varphi)}$
U_1-I_2	0	0	U_1-A_2	1	$e^{-\tau(T_2+\varphi)}$	I_1-A_2	1	$e^{-\tau(T_2+\varphi)}$
U_1-I_3	1	$e^{-\tau(T_3+\varphi)}$	U_1-A_3	0	0	I_1-A_3	0	0
U_2-I_1	0	0	U_2-A_1	0	0	I_2-A_1	0	0
U_2-I_2	2	$e^{-\tau(T_0+\varphi)} + e^{-\tau(T_1+\varphi)}$	U_2-A_2	1	$e^{-\tau(T_0+\varphi)}$	I_2-A_2	1	$e^{-\tau(T_0+\varphi)}$
U_2-I_3	1	$e^{-\tau(T_2+\varphi)}$	U_2-A_3	2	$e^{-\tau(T_1+\varphi)} + e^{-\tau(T_2+\varphi)}$	I_2-A_3	1	$e^{-\tau(T_1+\varphi)}$
						I_3-A_1	1	$e^{-\tau(T_3+\varphi)}$
						I_3-A_2	0	0
						I_3-A_3	1	$e^{-\tau(T_2+\varphi)}$

temporal graph $\mathcal{G}_t$is shown in Table 3.

We have elaborated the process of constructing a temporal multi-dimensional graph by taking a 3-dimensional graph as an example. It is worth noting that our proposed method is generic and can be easily expanded to discretionary dimensional heterogeneous graphs. In general, the construction of a temporal multi-dimensional graph consists of the following three steps. At first, each multi-dimensional interaction behavior data is mapped onto its corresponding long-time window in accordance with the interaction time t. Then the intensity of each interaction behavior between two nodes is calculated by adopting diverse time decay functions. Finally, the weight of an edge accumulates the intensities on all direct paths between two nodes.

4.2. Temporal graph probabilistic spreading via multi-dimensional meta-paths

We have respectively interpreted the construction of temporal multi-dimensional graphs and bipartite graph-based probabilistic spreading model in Section 4.1 and Section 2.2. In this section, we propose a dimension-free temporal graph probabilistic spreading framework via multi-dimensional meta-paths. Given a target user U_i, our goal is to recommend favorable and personalized items for him/her. We first assign some certain amount of the initial resources to all the selected items and the tagged tags by the target user, denoted as $f_t(I)$ and $f_t(A)$ respectively. For the convenience and conciseness, we take those 2-dimensional/3-dimensional meta-paths as examples to elaborate on the temporal graph probabilistic spreading process.

4.2.1. 2-dimensional meta-path-based temporal graph probabilistic spreading

As shown in Fig. 8(b), with Item or Tag as the source node type and Item as the target node type, the 2-dimensional meta-path set includes three meta-paths: {Item→User→Item, Item→Tag→Item, Tag→Item}. Take Item→User→Item meta-path as an example, we elaborate the temporal graph probabilistic spreading based on it.

Obtaining an initial resource $f_t(I)$ from the target user, each item equally distributes the resource to neighboring users. The resource of U_i gained from all neighboring items is:

$$f_1'(U_i) = \sum_{\beta=1}^{n} \frac{r_{\beta i} f_t(I_\beta)}{K_1(I_\beta)}, \tag{11}$$

where $f_t(I_\beta)$ is the initial resource of I_β, $K_1(I_\beta) = \sum_{i=1}^{m} r_{\beta i}$ is the degree of I_β in the temporal spreading process from items to users.

Each user then re-transfers the attained resource back to all items he/she has collected. The final resource of I_α obtained from temporal probabilistic spreading based on Item→User→Item meta-path is:

$$f_1''(I_\alpha) = \sum_{i=1}^{m} \frac{r_{\alpha i} f_1'(U_i)}{K_1(U_i)} = \sum_{i=1}^{m} \frac{r_{\alpha i}}{K_1(U_i)} \sum_{\beta=1}^{n} \frac{r_{\beta i} f_t(I_\beta)}{K_1(I_\beta)}, \tag{12}$$

where $K_1(U_i) = \sum_{\beta=1}^{n} r_{\beta i}$ is the degree of U_i in the temporal spreading process from users to items.

Finally, all U_i's unselected items are sorted in the descending order of $f_1''(I_\alpha)$, and the top L items are to be recommended to U_i as a recommendation list, denoted as $\mathcal{L}_{\mathrm{IUI}}^{U_i}$.

Analogously, the final resource of I_α obtained from temporal probabilistic spreading based on Item→Tag→Item meta-path is:

$$f_2''(I_\alpha) = \sum_{o=1}^{k} \frac{r_{\alpha o}}{K_1(A_o)} \sum_{\beta=1}^{n} \frac{r_{\beta o} f_t(I_\beta)}{K_2(I_\beta)}, \tag{13}$$

where $K_2(I_\beta) = \sum_{o=1}^{k} r_{\beta o}$ is the degree of I_β in the temporal spreading process from items to tags, while $K_1(A_o) = \sum_{\beta=1}^{n} r_{\beta o}$ is the degree of A_o in the temporal spreading process from tags to items.

All U_i's unselected items are sorted in the descending order of $f_2''(I_\alpha)$, and the top L items are to be recommended to U_i as a

recommendation list, denoted as $\mathscr{L}_{\mathrm{ITI}}{}^{U_i}$.

The final resource of I_α obtained from temporal probabilistic spreading based on Tag→Item meta-path is:

$$f_3''(I_\alpha) = \sum_{o=1}^{k} \frac{r_{\alpha o} f_{\mathrm{t}}(A_o)}{K_1(A_o)}, \tag{14}$$

where $f_{\mathrm{t}}(A_o)$ is the initial resource of A_o.

All U_i's unselected items are sorted in the descending order of $f_3''(I_\alpha)$, and the top L items are to be recommended to U_i as a recommendation list, denoted as $\mathscr{L}_{\mathrm{TI}}{}^{U_i}$.

4.2.2. 3-dimensional meta-path-based temporal graph probabilistic spreading

With Item or Tag as the source node type and Item node as the target type, the 3-dimensional meta-path set includes three meta-paths: {Item→User→Tag→Item, Item→Tag→User→Item, Tag→User→Item}. Take Item→User→Tag→Item meta-path as an example, we elaborate the temporal graph probabilistic spreading based on it.

Obtained an initial resource $f_{\mathrm{t}}(I)$ from the target user, each item equally distributes the resource to neighboring users. The resource of U_i gained from all neighboring items is:

$$f_1'(U_i) = \sum_{\beta=1}^{n} \frac{r_{\beta i} f_{\mathrm{t}}(I_\beta)}{K_1(I_\beta)}. \tag{15}$$

Each user then transfers the attained resource to all tags he/she has tagged. The resource of A_o obtained from all its neighboring users is:

$$f_4''(A_o) = \sum_{i=1}^{m} \frac{r_{io} f_1'(U_i)}{K_2(U_i)} = \sum_{i=1}^{m} \frac{r_{i0}}{K_2(U_i)} \sum_{\beta=1}^{n} \frac{r_{\beta i} f_{\mathrm{t}}(I_\beta)}{K_1(I_\beta)}, \tag{16}$$

where $K_2(U_i) = \sum_{o=1}^{k} r_{io}$ is the degree of U_i in the temporal spreading process from users to tags.

Subsequently, each tag re-distributes the acquired resource back to all items to which it has been assigned. The final resource of I_α obtained from temporal probabilistic spreading based on Item→User→Tag→Item meta-path is:

$$f_4''(I_\alpha) = \sum_{o=1}^{k} \frac{r_{\alpha o} f_4''(A_o)}{K_1(A_o)} = \sum_{o=1}^{k} \frac{r_{\alpha o}}{K_1(A_o)} \sum_{i=1}^{m} \frac{r_{io}}{K_2(U_i)} \sum_{\beta=1}^{n} \frac{r_{\beta i} f_{\mathrm{t}}(I_\beta)}{K_1(I_\beta)}. \tag{17}$$

All U_i's unselected items are sorted in the descending order of $f_4''(I_\alpha)$, and the top L items are to be recommended to U_i as a recommendation list, denoted as $\mathscr{L}_{\mathrm{IUTI}}{}^{U_i}$.

Analogously, the final resource of I_α obtained from temporal probabilistic spreading based on Item→Tag→User→Item meta-path is:

$$f_5''(I_\alpha) = \sum_{i=1}^{m} \frac{r_{\alpha i}}{K_1(U_i)} \sum_{o=1}^{k} \frac{r_{io}}{K_2(A_o)} \sum_{\beta=1}^{n} \frac{r_{\beta o} f_{\mathrm{t}}(I_\beta)}{K_2(I_\beta)}, \tag{18}$$

where $K_2(A_o) = \sum_{i=1}^{m} r_{io}$ is the degree of A_o in the temporal spreading process from tags to users.

All U_i's unselected items are sorted in the descending order of $f_5''(I_\alpha)$, and the top L items are to be recommended to U_i as a recommendation list, denoted as $\mathscr{L}_{\mathrm{ITUI}}{}^{U_i}$.

The final resource of I_α obtained from temporal probabilistic spreading based on Tag→User→Item meta-path is:

$$f_6''(I_\alpha) = \sum_{i=1}^{m} \frac{r_{\alpha i}}{K_1(U_i)} \sum_{o=1}^{k} \frac{r_{io} f_{\mathrm{t}}(A_o)}{K_2(A_o)}. \tag{19}$$

All U_i's unselected items are sorted in the descending order of $f_6''(I_\alpha)$, and the top L items are to be recommended to U_i as a recommendation list, denoted as $\mathscr{L}_{\mathrm{TUI}}{}^{U_i}$.

It is worth emphasizing that this spreading approach is generic, dimension-free, and task-independent. It makes it easy to fulfill temporal probabilistic spreading based on N-dimensional meta-paths by continually expanding the added node types to generate N-dimensional meta-paths. Moreover, the proposed temporal graph probabilistic spreading framework via multi-dimensional meta-paths can also be used for other recommendation tasks, e.g. recommending tags to a user or an item, items to a tag, users to an item or a tag etc.

4.3. Boosting strategies for personalized recommendation

Each user has diverse and personalized preferences in accordance with some preferred special meta-paths. To capture these preferences, we have proposed temporal graph probabilistic spreading framework via diverse multi-dimensional meta-paths in Section 4.2. How to utilize the framework for personalized recommendation is crucial yet challenging. It is time-consuming to learn the

Table 4
Statistics of datasets. #Users, #Items, and #Tags refer to the number of users, items (movies in MovieLens and artists in Last.fm), and tags respectively. #Edges refers to the number of interactions among users, items, and tags. Sparsity indicates the sparsity degree of the datasets, which is equal to #Edges/(#Users × #Items × #Tags). $t_{\min}$ and $t_{\max}$ represent the minimum and the maximum timestamp.

Dataset	#Users	#Items	#Tags	#Edges	Sparsity	$t_{\min}$	$t_{\max}$
MovieLens	2113	5908	9079	47957	4.23×10^{-7}	1135361580	1231108633
Last.fm	1892	12523	9748	186479	8.07×10^{-7}	1122847200	1304941497

weights of all users' preferences on every meta-path by utilizing network embedding/represent learning methods. To address this issue, in this paper, we propose two suitable boosting strategies to automatically learn the priority and importance of these multi-dimensional meta-paths at the user-level granularity and to make desirable and personalized recommendation for each individual user.

In Algorithm 1, each temporal graph probabilistic spreading via a certain meta-path is casted as a fundamental learning algorithm. We propose a vote-accumulation mechanism to automatically acquire the recommended priority and weighted balance to boost the recommendation quality. For example, given a target user U_i, the list set of the top L recommended items via multi-dimensional meta-paths temporal graph probabilistic spreading, denoted as $\mathscr{L}_{\mathscr{P}}{}^{U_i} = \{\mathscr{L}_{\mathrm{IUI}}{}^{U_i}, \mathscr{L}_{\mathrm{ITI}}{}^{U_i}, \mathscr{L}_{\mathrm{TI}}{}^{U_i}, \mathscr{L}_{\mathrm{IUTI}}{}^{U_i}, \mathscr{L}_{\mathrm{ITUI}}{}^{U_i}, \mathscr{L}_{\mathrm{TUI}}{}^{U_i}, \ldots\}$, reflects U_i's preferences on multi-dimensional meta-paths. Line 5 in Algorithm 1 is to add 1 to the number of vote of an item if it has been voted by a fundamental learning algorithm. Analogously, Line 6 is to accumulate the resource of this item. Line 13 is to sort $\mathscr{L}^{U_i}$ in the descending order of the number of items' votes and then in descending order of items' resources. Finally, the top L items in the sorted $\mathscr{L}^{U_i}$ are to be recommended to the target user U_i.

Different from Algorithm 1, Algorithm 2 adopts a maximum-rearrangement mechanism, which first maximizes the resource of each item in the recommendation list set $\mathscr{L}_{\mathscr{P}}{}^{U_i}$ derived from diverse temporal graph probabilistic spreading methods via multi-dimensional meta-paths, and then rearranges $\mathscr{L}^{U_i}$ in the descending order of items' resources.

5. Experimental process and results

To demonstrate the effectiveness and superiority of our proposed MD-MP-TGPS method, this study adopts two golden real-world datasets, hetrec2011-movielens-2k and hetrec2011-lastfm-2k, released in the top conference HetRec 2011 (http:/ir.ii.uam.es/hetrec2011) in the field of recommender systems by a professional team of experts from the Universidad Autónoma de Madrid. We comprehensively compare MD-MP-TGPS with eight other state-of-the-art algorithms in accuracy, diversity, and novelty based on performance indicators (namely precision, recall, F1, popularity, hamming distance along with intra-list diversity).

5.1. Description of experimental datasets

Hetrec2011-movielens-2k dataset is an extension of MovieLens10M dataset released by GroupLens research group and hetrec2011-lastfm-2k dataset originates from an online website Last.fm (https://www.last.fm/) which provides the services of playing music, finding songs, and finding out favorite artists. For brevity, we will refer to these two datasets as MovieLens and Last.fm. To construct multi-dimensional temporal graphs, we respectively utilize the three-way interaction data among users, movies and tags derived from the "user_taggedmovies-timestamps.dat" file in the MovieLens dataset and the three-way interaction data among users, artists and tags originated from the "user_taggedartists-timestamps.dat" file in the Last.fm dataset to construct 3-dimensional temporal graphs. The statistics of these datasets are described in Table 4.

Following the work (Wang & Han, 2021), we adopt the temporal partition method and divide the training set and the probe set in strict chronological order. In MovieLens dataset, we evenly divide the data into 24 subsets with a time length of 45 days in each subset, denoted as $\mathrm{D}=\{\mathrm{D}_1, \mathrm{D}_2, \ldots, \mathrm{D}_{24}\}$. Thereafter, we slide the subsets to generate the training set and its corresponding probe set. After five chronological sliding, five pairs of training sets and probe sets are formed, as illustrated in: $\{\mathrm{ET}_1=\bigcup_{i=1}^{19} D_i, \mathrm{EP}_1=\mathrm{D}_{20}\}$, $\{\mathrm{ET}_2=\bigcup_{i=1}^{20} D_i, \mathrm{EP}_2=\mathrm{D}_{21}\}$, $\{\mathrm{ET}_3=\bigcup_{i=1}^{21} D_i, \mathrm{EP}_3=\mathrm{D}_{22}\}$, $\{\mathrm{ET}_4=\bigcup_{i=1}^{22} D_i, \mathrm{EP}_4=\mathrm{D}_{23}\}$, $\{\mathrm{ET}_5=\bigcup_{i=1}^{23} D_i, \mathrm{EP}_5=\mathrm{D}_{24}\}$ where $\cup$ is the union operator. Analogously, in Last.fm dataset, we evenly divide the data into 35 subsets with a time length of 60 days in each subset. Each group of experiments in this paper are conducted on these five groups of training sets and probe sets. Finally, the average value of all these experiments is set as the final experimental result.

It is worth mentioning that cold users may appear along with data temporal division, that is, users are present in the probe set but not in the training set. Adopting a preprocessing method to filter out those cold users, this paper mainly focuses on the aspects of accuracy, diversity, and novelty of recommendations using multi-dimensional meta-paths temporal graph probability spreading. The cold start problem is indeed an interesting, important and challenging issue in recommender systems and we will study this interesting topic in our future work.

5.2. Performance indicators

To investigate the recommendation quality of the proposed MD-MP-TGPS in contrast to the compared methods, this paper comprehensively measures the aspects of accuracy, diversity, and novelty of recommendation by employing six widely-used metrics: precision, recall, F1, hamming distance, intra-list diversity, and popularity.

(1) Precision (P) (Wang & Han, 2021): Precision refers to the proportion of items in the recommendation list that are actually selected by the target user. The average precision means the recommendation precision on average over all users, denoted as

$$\mathrm{P}(L) = \frac{1}{m}\sum_{i=1}^{m}\mathrm{P}_i(L) = \frac{1}{m}\sum_{i=1}^{m}\frac{\mathrm{O}_i}{L}, \tag{20}$$

where O_i is the number of items in the recommendation list that are actually selected by U_i.

Higher precision indicates better recommendation accuracy.

(2) Recall (R) (Wang & Han, 2021): Recall refers to the proportion of items in the probe set located in the recommendation list.

The average recall signifies the recommendation recall on average over all users, defined as

$$\mathrm{R}(L) = \frac{1}{m}\sum_{i=1}^{m}\mathrm{R}_i(L) = \frac{1}{m}\sum_{i=1}^{m}\frac{\mathrm{O}_i}{|\mathrm{PS}_i|}, \tag{21}$$

where $|\mathrm{PS}_i|$ is the number of items in the probe set that are actually selected by U_i.

Higher recall proves better recommendation accuracy.

(3) F1-score (F1) (Wang & Han, 2021): F1-score refers to the harmonic average of precision and recall, which is computed as

$$\mathrm{F1}(L) = \frac{2\cdot\mathrm{P}(L)\cdot\mathrm{R}(L)}{\mathrm{P}(L)+\mathrm{R}(L)}. \tag{22}$$

Higher F1-score indicates better recommendation accuracy.

(4) Hamming distance (H) (Wang & Han, 2021, Zhou et al., 2010): With the consideration of inter-list diversity, hamming distance focuses on the difference of recommendation lists provided by a recommender system to two peer users. The average hamming distance is defined as

$$\mathrm{H}(L) = \frac{1}{m(m-1)}\sum_{i=1}^{m}\sum_{j=1,j\neq i}^{m}1-\frac{\mathrm{Q}_{ij}(L)}{L}, \tag{23}$$

where $\mathrm{Q}_{ij}(L)$ is the number of common items existing in both the recommendation list to U_i and that to U_j.

Higher hamming distance means better recommendation diversity.

(5) Intra-list diversity (ILD) (Castells, Hurley, & Vargas, 2015): Unlike hamming distance, which focuses on inter-list diversity, ILD concerns the degree of diversity of recommended items within the same recommendation list. The average intra-list diversity is defined as

$$\mathrm{ILD}(L) = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{L(L-1)}\sum_{I_\alpha\in\mathscr{L}^{U_i}(L)}\sum_{I_\beta\in\mathscr{L}^{U_i}(L)}\left(1-\mathrm{Sim}(I_\alpha,I_\beta)\right)\right), \tag{24}$$

where $\mathscr{L}^{U_i}(L)$ is the recommendation list to U_i, $\mathrm{Sim}(I_\alpha, I_\beta)$ is a similarity/relevance measure function, which represents the similarity/relevance of I_α and I_β. In this paper, we adopt the Jaccard similarity as the similarity measure function. Suppose that $\mathrm{U}(I_\alpha)$, $\mathrm{U}(I_\beta)$ are the set of users that have interacted with I_α and I_β, respectively. The Jaccard similarity $\mathrm{Sim}(I_\alpha, I_\beta)$ is then defined as

$$\mathrm{Sim}(I_\alpha,I_\beta) = \frac{\left|\mathrm{U}(I_\alpha)\cap\mathrm{U}(I_\beta)\right|}{\left|\mathrm{U}(I_\alpha)\cup\mathrm{U}(I_\beta)\right|}. \tag{25}$$

Higher intra-list diversity means better recommendation diversity.

(6) Popularity (PO) (Wang & Han, 2021; Cami, Hassanpour, & Mashayekhi, 2019): Popularity quantifies the ability of a recommender system to provide novel and serendipitous items. The average popularity is define as

$$\mathrm{PO}(L) = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{L}\sum_{I_\alpha\in\mathscr{L}^{U_i}(L)}K(I_\alpha)\right) = \frac{1}{mL}\sum_{i=1}^{m}\sum_{I_\alpha\in\mathscr{L}^{U_i}(L)}K(I_\alpha), \tag{26}$$

where $K(I_\alpha)$ is the degree of I_α, and $\mathscr{L}^{U_i}(L)$ is the recommendation list to U_i.

Lower popularity implies better recommendation novelty.

5.3. Benchmark methods and parameters setting

To consolidate the effectiveness and credibility of the proposed MD-MP-TGPS method, we comprehensively compare it with eight other state-of-the-art baselines, namely 2-dimensional random recommendation(2D-Random), 2-dimensional popularity-based recommendation(2D-TP), 2-dimensional meta-path-based graph spreading recommendation(2D-ProbS), 2-dimensional Markov-based temporal recommendation (2D-Markov), 2-dimensional time decay-based temporal recommendation (2D-NBIt), 3-dimensional meta-path-based graph spreading recommendation(3D-IUAI, 3D-IAUI), and 3-dimensional meta-path-based hybrid recommendation (3D-IUI-IAI). These compared baselines include not only simple ones like random recommendation and popularity-based recommendation but also 2-dimensional/3-dimensional/hybrid meta-path-based recommendation and temporal recommendation.

(1) 2D-Random: It is a random recommendation baseline. Given a target user U_i and the length of recommendation list L, 2D-Random randomly selects L items from the set of items not selected by the user U_i and recommends them to the user U_i. To avoid the volatility of a single randomized sampling, we conduct 50 experiments and take the average value as the final result of the 2D-Random method.
(2) 2D-TP: It is a popularity-based baseline. Total popularity, denoted as $K(I_\alpha)$, represents the degree of I_α. Given a target user U_i and the length of recommendation list L, 2D-TP selects L items with the largest total popularity $K(I_\alpha)$ from the set of items not selected by U_i and recommends them to U_i as a recommendation list.
(3) 2D-ProbS: As a benchmark for evaluating performance of recommendation algorithms, the classic bipartite graph-based recommendation approach, ProbS, is commonly used for its high recommendation accuracy Zhou et al., 2010). As described in Section 3.2, given a target user U_i, after calculating the final resource with Eqs. (1)-((2), all U_i's unselected items are sorted in the descending order of the final resource, and the top L items are to be recommended to U_i as a recommendation list.
(4) 2D-Markov: Markov-based algorithms are widely used in sequential prediction/temporal recommendation and graph traversal/random walk (Johnson & Ng, 2017). Using a Markov process, a graph (here a user-item bipartite graph) achieves a stationary distribution state and the according convergent state matrix represents the probabilities of a user's preference for an item after several rounds of iteration. Then the top L uncollected items with maximum probability are to be commended to the target user.
(5) 2D-NBIt: As an effective recommendation strategy, the time decay-based recommendation is commonly used as a baseline for the performance measurement of temporal recommender systems. NBIt is a temporal recommendation method adopting a nonlinear exponential decay function (Wang & Han, 2020).
(6) 3D-IUAI: Poulain and Tarissan (2020) composited a user-product-category tripartite graph and adopted a random walk based on 3-dimensional meta-path category→product→user. Huang, Chen, Wang, and Mei (2014) built a user-query-video tripartite graph and leveraged the graph propagation based on 3-dimensional meta-path video→query→user for personalized video recommendation. Following the work (Poulain & Tarissan, 2020; Huang, Chen, Wang, & Mei, 2014), different from 2-dimensional meta-path-based methods 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI is a graph probabilistic spreading method via the 3-dimensional meta-path Item→User→Tag→Item.
(7) 3D-IAUI: Following the work (Poulain & Tarissan, 2020; Huang, Chen, Wang, & Mei, 2014), similar to 3D-IUAI, 3D-IAUI is another graph probabilistic spreading method via 3-dimensional meta-path Item→Tag→User→Item.
(8) 3D-IUI-IAI: Following the work (Zhang, Zhou, & Zhang, 2010; Li, Tang, & Chen, 2017), different from 3D-IUAI and 3D-IAUI, 3D-IUI-IAI incorporates recommendation results via two 2-dimensional meta-paths Item→User→Item and Item→Tag→Item by a tunable parameter λ.
(9) MD-MP-TGPS1: It is a concrete implementation of our proposed MD-MP-TGPS method. At the personalized recommendation stage, it applies a boosting strategy of the vote-accumulation mechanism represented in the Algorithm 1.
(10) MD-MP-TGPS2: It is another concrete implementation of our proposed MD-MP-TGPS method. Different from MD-MP-TGPS1, it applies a boosting strategy of the maximum-rearrangement mechanism represented in the Algorithm 2 at the personalized recommendation stage.

To ensure all algorithms perform under uniform experimental conditions, all the methods in this paper, including our proposed MD-MP-TGPS and the compared ones, are implemented and carried out in the same framework and platform with an Intel Xeon Silver 4114 2.20GHz CPU and 256GB of RAM running Python 3.7.0, and the server operating system is CentOS Linux release 7.4.1708. The source code of all algorithms will be released at https://github.com/wangyang-aqnu/MD-MP-TGPS. The parameter setting of these algorithms is described as follows:

For 2D-TP, 2D-ProbS, 3D-IUAI, and 3D-IAUI, there are no parameters to set in these algorithms.

For 2D-Random, although there are no parameters to be tuned in the algorithm itself, to avoid the volatility of a randomized experiment on the recommendation result, we conducted 50 2D-Random experiments and taken the average value as the experimental result of the algorithm.

For 2D-Markov, to converge the Markov process to a stable state, we set the number of iterations to 30.

For 3D-IUI-IAI, it incorporates recommendation results of 2D-IUI (via 2-dimensional meta-path Item→User→Item) and 2D-IAI (via 2-dimensional meta-path Item→Tag→Item) by a tunable parameter $\lambda \in [0, 1]$. In this paper, to make both meta paths equally important, we set λ to 0.5.

For 2D-NBIt, MD-MP-TGPS1, and MD-MP-TGPS2, these algorithms all adopted a nonlinear exponential time decay function based on Newton's law of cooling $r_{\alpha i}{}^T = a_{\alpha i} t e^{-\tau(T+\varphi)}$ (Wang & Han, 2020). The function includes two parameters: decay rate τ and bias offset

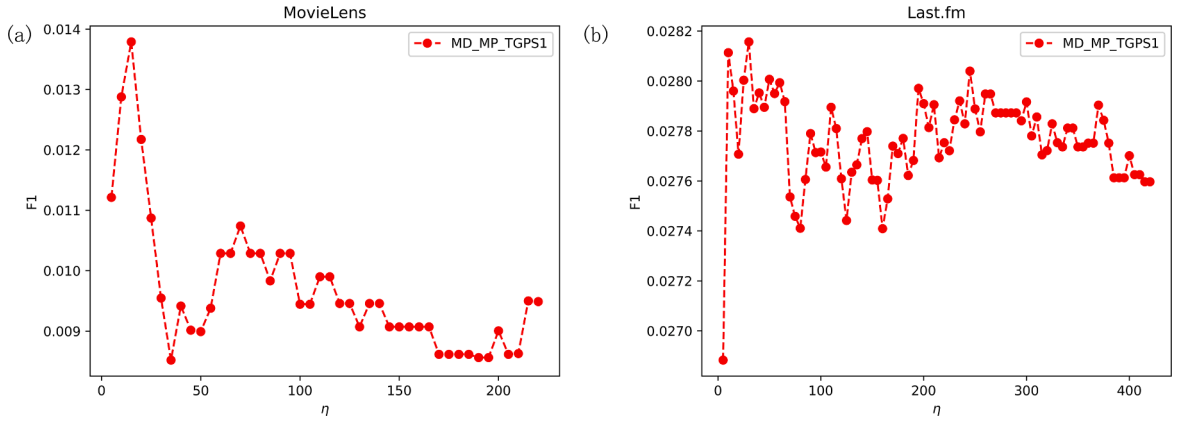


Fig. 9. The average F1-score (F1) of MD-MP-TGPS1 versus the long-time window parameter (η) with the length of recommendation list L=10 in (a) MovieLens and (b) Last.fm.

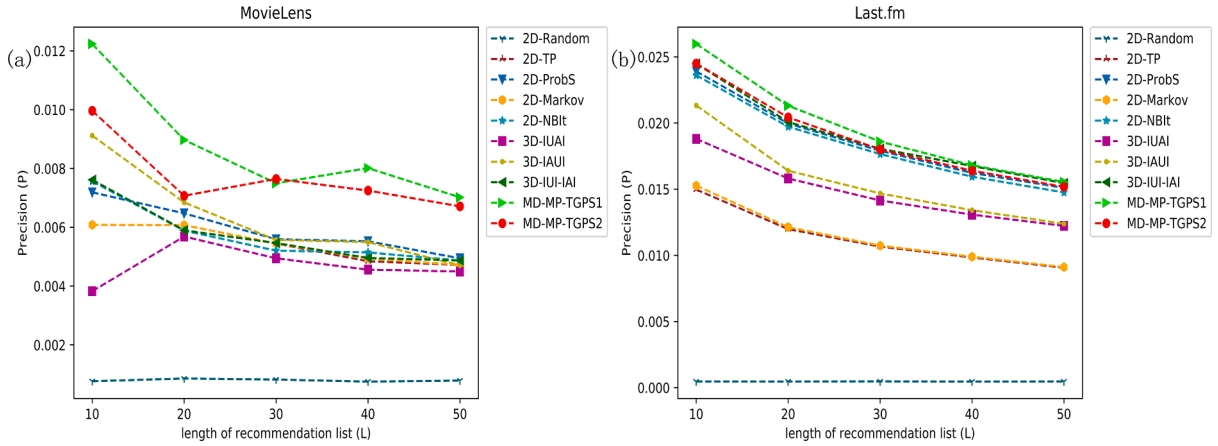


Fig. 10. Performance comparison of MD-MP-TGPS (MD-MP-TGPS1, MD-MP-TGPS2) with other diversified baselines (2D-Random, 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, and 3D-IUI-IAI) in varied recommendation lengths in precision in (a) MovieLens and (b) Last.fm.

φ. Based on the work (Wang & Han, 2020), on the MovieLens dataset, we set $r_{\alpha i}{}^{t}(init) = a_{\alpha i}{}^{t}$, $r_{\alpha i}{}^{t}(finish) = 0.25\ a_{\alpha i}{}^{t}$, and g=24. It means that attenuation starts from the initial value $a_{\alpha i}{}^{t}$ and $a_{\alpha i}{}^{t}$ attenuates to the final value $0.25a_{\alpha i}{}^{t}$ through 24 long-time windows. Plugging $a_{\alpha i}{}^{t}(init) = a_{\alpha i}{}^{t}$, $r_{\alpha i}{}^{t}(finish) = 0.25\ a_{\alpha i}{}^{t}$, and g=24 into the Eq (27), we get τ≈0.0577623, and φ=0.

$$\begin{cases} \tau = \dfrac{1}{g}\ln\dfrac{r_{\alpha i}{}^{t}(init)}{r_{\alpha i}{}^{t}(finish)}, \\ \varphi = \dfrac{1}{\tau}\ln\dfrac{r_{\alpha i}{}^{t}}{r_{\alpha i}{}^{t}(init)}. \end{cases} \tag{27}$$

Analogously, in Last.fm, we set $a_{\alpha i}{}^{t}(init) = a_{\alpha i}{}^{t}$, $r_{\alpha i}{}^{t}(finish) = 0.25\ a_{\alpha i}{}^{t}$, and g=35. After a similar calculation, we get τ≈0.0396084, and φ=0.

Besides τ and φ, 2D-NBIt, MD-MP-TGPS1, and MD-MP-TGPS2 also include a long time-window parameter η. We conducted two groups of experiments in MovieLens and Last.fm datasets respectively to measure F1-score of MD-MP-TGPS1 as a function of η with the length of recommendation list L being 10. In MovieLens, η varies from 5 to 220 with the step size set as 5. In Last.fm, η varies from 5 to 420 with the step size being 5. Experimental results are shown in Fig. 9. To balance the accuracy and diversity of recommendation and ensure the fair comparison, we set the optimal value of the long time-window parameter η^* =15 in MovieLens and η^* =30 in Last.fm for 2D-NBIt, MD-MP-TGPS1, and MD-MP-TGPS2.

Besides τ, φ, and η, 2D-NBIt has a spreading probability parameter δ, whose function is to inhibit the recommendation intensity of high-degree nodes, with the value range being [0, 1]. δ=1 means that the degrees of all nodes in temporal graphs are not suppressed while δ=0 denotes that the degrees of all nodes are suppressed to 1. Since the other algorithms do not suppress the degree of nodes, in this paper, we set δ=1 in order to make a fair comparison with these algorithms.

Table 5
Precision comparison among 2D-Random, 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, 3D-IUI-IAI, MD-MP-TGPS1, and MD-MP-TGPS2. Column L refers to the length of recommendation list. Avg calculates the average value of experimental results with L being 10, 20, 30, 40 and 50.

Datasets	L	2D-Random	2D-TP	2D-ProbS	2D-Markov	2D-NBIt	3D-IUAI	3D-IAUI	3D-IUI-IAI	MD-MP-TGPS1	MD-MP-TGPS2
MovieLens	10	0.000763	0.006082	0.007188	0.006082	0.008712	0.003825	0.009117	0.007615	0.012234	0.009962
	20	0.000854	0.006071	0.006472	0.006071	0.008	0.005684	0.006845	0.005895	0.00897	0.007072
	Avg	0.000794	0.005432	0.005943	0.005451	0.00752	0.004701	0.006345	0.005759	**0.008746**	0.007727
Last.fm	10	0.000465	0.014969	0.023909	0.01527	0.023611	0.018803	0.021329	0.024489	0.025976	0.024494
	20	0.00046	0.01198	0.019965	0.012129	0.01972	0.015808	0.016396	0.020074	0.021309	0.020426
	Avg	0.000463	0.011302	0.018622	0.011427	0.018335	0.014811	0.015639	0.018964	**0.019645**	0.018898

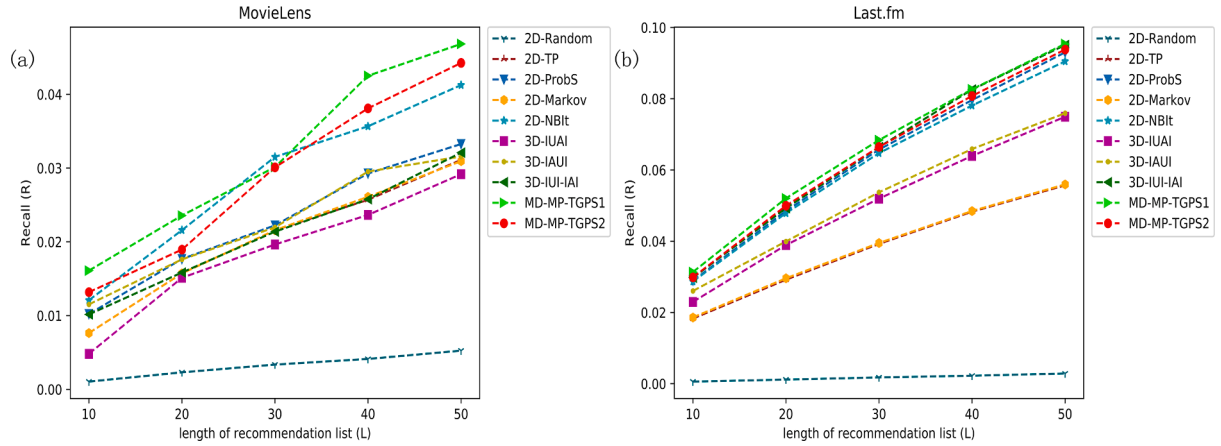


Fig. 11. Performance comparison of MD-MP-TGPS with other diversified baselines in varied recommendation lengths in recall in (a) MovieLens and (b) Last.fm.

5.4. Performance comparison and experimental results

We comprehensively compare our proposed MD-MP-TGPS method with eight other state-of-the-art baselines, namely 2D-Random, 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, and 3D-IUI-IAI. Experimental results differ significantly in performance in accuracy evaluation metrics of precision, recall, and F1-score, in diversity evaluation metric of hamming distance and intra-list diversity, and in novelty evaluation metric of popularity, as shown in Figs. 10-15 and Table 5-10.

5.4.1. Precision

Shown in Fig. 10 and Table 5 is the precision of recommendation of both the proposed MD-MP-TGPS method and the compared ones in MovieLens and Last.fm datasets. As illustrated, compared with the other methods, MD-MP-TGPS displays better precision.

Regarding to precision in MovieLens, both our proposed MD-MP-TGPS1 and MD-MP-TGPS2 surpass significantly all the compared methods, namely 2D-Random, 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, and 3D-IUI-IAI. Particularly, MD-MP-TGPS1 achieves the best performance with its average precision up by 1001.51% compared with 2D-Random, by 61.01% compared with 2D-TP, by 47.16% with 2D-ProbS, by 60.45% with 2D-Markov, by 16.30% with 2D-NBIt, by 86.05% with 3D-IUAI, by 37.84% with 3D-IAUI, and by 51.87% with 3D-IUI-IAI. Among the diverse compared methods, 2D-NBIt attains better precision than other methods.

As to precision in Last.fm, MD-MP-TGPS1 and MD-MP-TGPS2 also present better performance than such compared methods as 2D-Random, 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, and 3D-IUI-IAI. The average precision of MD-MP-TGPS1 is increased by 4142.98% compared with 2D-Random, by 73.82% with 2D-TP, by 5.49% with 2D-ProbS, by 71.92% with 2D-Markov, by 7.14% with 2D-NBIt, by 32.64% with 3D-IUAI, by 25.62% with 3D-IAUI, and by 3.59% with 3D-IUI-IAI. The precisions of 3D-IUI-IAI, 2D-ProbS, and 2D-NBIt, amongst all the compared methods, indicate their superiority over the pure 3-dimensional meta-path-based methods 3D-IUAI and 3D-IUAI. In particular, 3D-IUI-IAI maintains better performance and ranks second in the metric of precision, just next to MD-MP-TGPS1.

5.4.2. Recall

Similar to the metric precision, recall presents the similar result. What is illustrated in Fig. 11 and Table 6 is the recall comparison among our suggested method and those compared ones in some typical lengths in MovieLens and Last.fm datasets. It can be concluded that the proposed method demonstrates higher recall, suggesting more excellence in capturing the actual preferences of users than all the other compared methods.

In recall comparison in MovieLens, MD-MP-TGPS1 and MD-MP-TGPS2 significantly outperform 2D-Random, 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, and 3D-IUI-IAI. MD-MP-TGPS1 still achieves the best performance. To be specific, the average recall is raised by 12% against that of 2D-NBIt, by 56.44%, 41.30%, 56.09%, 72.27%, 41.85%, 51.25%, and 892.17% with that of 2D-TP, 2D-Probs, 2D-Markov, 3D-IUAI, 3D-IAUI, 3D-IUI-IAI, and 2D-Random respectively. Amongst all the compared methods, 2D-NBIt obtains better performance than other methods in the recall metric.

In Last.fm, the contrast of recall confirms that MD-MP-TGPS1 and MD-MP-TGPS2 also acquire better performance than other state-of-the-art methods, with the average recall up by 3776.78%, 72.98%, 4.43%, 71.71%, 6.43%, 30.52%, 26.12% and 2.08% respectively when compared with that of 2D-Random, 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IUAI, and 3D-IUI-IAI. The recall of 3D-IUI-IAI, 2D-ProbS, and 2D-NBIt, amongst the contrasted methods, proves to be superior to that of 3D-IUAI, 3D-IAUI, 2D-TP, and 2D-Markov. Strikingly, 3D-IUI-IAI works well, ranking second to MD-MP-TGPS1. It shows that the method of fusing two 2-dimension graph spreading models is quite effective in this dataset.

Table 6
Recall comparison among 2D-Random, 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, 3D-IUI-IAI, MD-MP-TGPS1, and MD-MP-TGPS2. Column *L* refers to the length of recommendation list. Avg calculates the average value of experimental results with *L* set at 10, 20, 30, 40 and 50.

Datasets	L	2D-Random	2D-TP	2D-ProbS	2D-Markov	2D-NBIt	3D-IUAI	3D-IAUI	3D-IUI-IAI	MD-MP-TGPS1	MD-MP-TGPS2
MovieLens	10	0.001041	0.007624	0.010201	0.007624	0.012072	0.004808	0.011506	0.010149	0.016086	0.013164
	20	0.002295	0.015712	0.017664	0.015712	0.021571	0.015115	0.017611	0.01583	0.023558	0.018935
	Avg	0.003207	0.020339	0.022518	0.020385	0.028409	0.01847	0.022431	0.021038	**0.031819**	0.028905
Last.fm	10	0.000571	0.018253	0.028895	0.018587	0.028666	0.022935	0.026016	0.029721	0.031329	0.02983
	20	0.001133	0.029179	0.048517	0.02957	0.047866	0.038899	0.03992	0.049255	0.052023	0.049946
	Avg	0.001701	0.038123	0.063146	0.038405	0.061961	0.050524	0.052285	0.064599	**0.065944**	0.064154

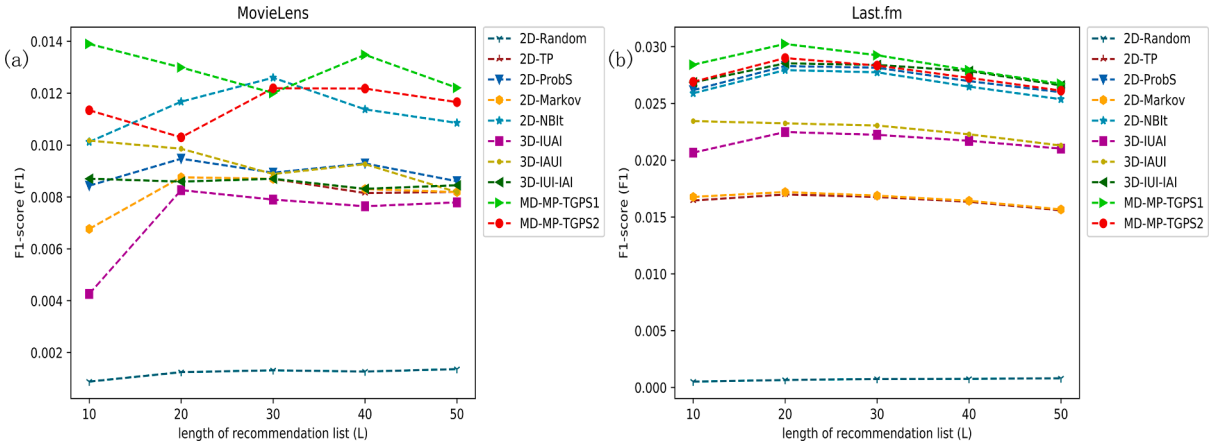


Fig. 12. Performance comparison of MD-MP-TGPS with other diversified baselines in varied recommendation lengths in the F1-score metric in (a) MovieLens and (b) Last.fm.

5.4.3. F1-score

When compared with other methods, the proposed approach shows better F1-score, indicating the same excellence as shown in the performance of precise and recall. What is depicted in Fig. 12 and Table 7 is the average F1-score and F1-score in some typical lengths in MovieLens and Last.fm datasets, signifying more stability in our recommendation method than those compared ones when applied in different datasets.

In MovieLens, the F1-score comparison suggests that MD-MP-TGPS1 and MD-MP-TGPS2 significantly outperform 2D-Random, 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, and 3D-IUI-IAI. MD-MP-TGPS1 proves the best F1-score, with the average being improved by 963.13%, 59.21%, 44.36%, 58.65%, 14.08%, 80.15%, 39.30%, and 51.06% when compared with 2D-Random, 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, and 3D-IUI-IAI respectively. Among the compared algorithms, 2D-NBIt displays the best recommendation performance.

In Last.fm, the F1-score comparison shows that MD-MP-TGPS1 and MD-MP-TGPS2 also exceed the equivalent performance of those compared baselines. Take the average F1-score of MD-MP-TGPS1 for example, it is increased quite a lot, namely by 4013.42% when compared with that of 2D-Random, by 73.54% with 2D-TP, by 5.16% with 2D-ProbS, by 71.84% with 2D-Markov, by 6.84% with 2D-NBIt, by 31.84% with 3D-IUAI, by 25.78% with 3D-IAUI, and by 3.13% with 3D-IUI-IAI. Amongst those compared methods, the hybrid method 3D-IUI-IAI presents the best performance in the F1-score metric, ranking next to MD-MP-TGPS1. The F1-score of 3D-IUI-IAI, 2D-ProbS, and 2D-NBIt works better while that of 2D-Random, 2D-TP, 2D-Markov is worse. This suggests that it is not feasible to simply recommend hot/popular artists i Last.fm.

5.4.4. Hamming distance

The average hamming distance is presented along with that in some typical lengths in MovieLens and Last.fm datasets in Fig. 13 and Table 8. Although the random recommendation method (2D-Random) has good diversity and novelty, the recommendation accuracy is so poor that it can only be used as an auxiliary recommender for cold users. Therefore, we do not list 2D-Random in Table 8, nor in Table 9-10 below. Compared with other methods, the presented MD-MP-TGPS2 method displays better hamming distance, proving richer diversity in recommendation.

In average hamming distance, experimental results in MovieLens show an increase, compared with other methods, of 313.50% with 2D-TP, 12.82% with 2D-ProbS, 314.88% with 2D-Markov, 10.77% with 2D-NBIt, 5.75% with 3D-IUAI, 14.91% with 3D-IAUI, and 0.75% with 3D-IUI-IAI, to be specific. In Last.fm, the average hamming distance is promoted by 284.77% with 2D-TP, 2.31% with 2D-ProbS, 244.04% with 2D-Markov, 2.89% with 2D-NBIt, and 32.24% with 3D-IAUI respectively. Amid those compared recommendation methods, 3D-IUI-IAI and 3D-IUAI perform well and show excellent inter-list diversity.

5.4.5. Intra-list diversity

The average intra-list diversity is presented along with that in some typical lengths in MovieLens and Last.fm datasets in Fig. 14 and Table 9. Compared with other methods, the presented MD-MP-TGPS2 method displays better intra-list diversity.

Experimental results in MovieLens show a slight increase in average intra-list diversity, compared with other methods, 7.88% with 2D-TP, 4.12% with 2D-ProbS, 8.77% with 2D-Markov, 2.34% with 2D-NBIt, 0.74% with 3D-IUAI, 2.44% with 3D-IAUI, and 1.84% with 3D-IUI-IAI, to be specific. Similar promotion is proved in Last.fm, with the average intra-list diversity increased by 2.81% with 2D-TP, 2.21% with 2D-ProbS, by 4.68% with 2D-Markov, 1.85% with 2D-NBIt, 1.17% with 3D-IAUI, and 0.64% with 3D-IUI-IAI respectively. Amid those compared recommendation methods, 3D-IUAI performs well and shows excellent intra-list diversity.

5.4.6. Popularity

The popularity is compared in MovieLens and Last.fm datasets with the average displayed in Fig. 15 and Table 10 along with that in

Table 7
F1-score comparison among 2D-Random, 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, 3D-IUI-IAI, MD-MP-TGPS1, and MD-MP-TGPS2. Column *L* refers to the length of recommendation list. Avg calculates the average value of experimental results with *L* set at 10, 20, 30, 40 and 50.

Datasets	*L*	2D-Random	2D-TP	2D-ProbS	2D-Markov	2D-NBIt	3D-IUAI	3D-IAUI	3D-IUI-IAI	MD-MP-TGPS1	MD-MP-TGPS2
MovieLens	10	0.000881	0.006766	0.008433	0.006766	0.01012	0.004261	0.010173	0.008701	0.013898	0.011341
	20	0.001245	0.008758	0.009473	0.008758	0.011672	0.008262	0.009858	0.008591	0.012992	0.010298
	Avg	0.001215	0.008113	0.008948	0.008142	0.011323	0.00717	0.009273	0.008551	**0.012917**	0.011532
Last.fm	10	0.000513	0.016449	0.026167	0.016766	0.025894	0.020664	0.023441	0.026853	0.028403	0.0269
	20	0.000654	0.016986	0.028289	0.017202	0.027932	0.02248	0.023245	0.028523	0.030234	0.028994
	Avg	0.000693	0.016426	0.027108	0.016589	0.026681	0.021622	0.022664	0.02764	**0.028506**	0.02752

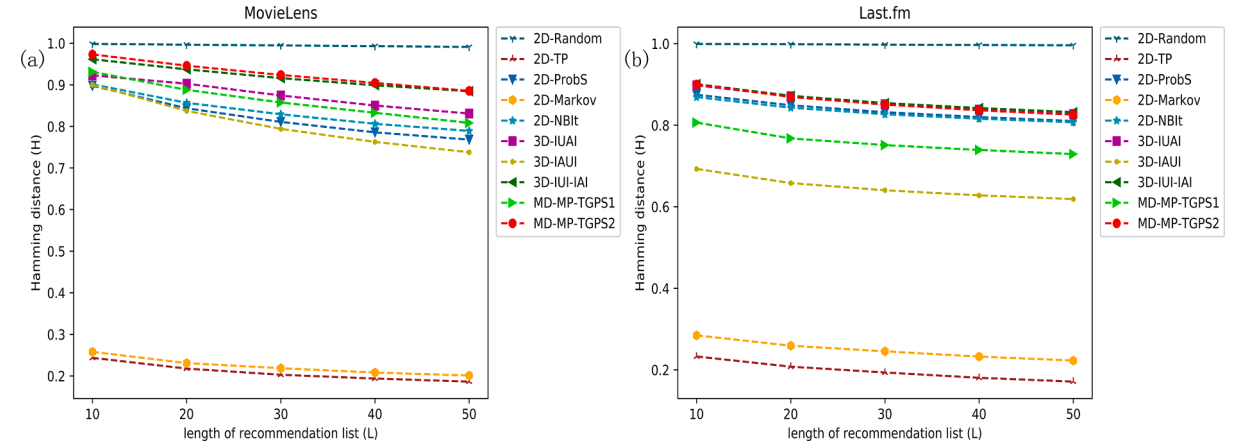


Fig. 13. Performance comparison of MD-MP-TGPS with other diversified baselines in varied recommendation lengths in hamming distance in (a) MovieLens and (b) Last.fm.

Table 8
Hamming distance comparison among 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, 3D-IUI-IAI, MD-MP-TGPS1, and MD-MP-TGPS2. Column *L* refers to the length of recommendation list. Avg calculates the average value of experimental results with *L* set at 10, 20, 30, 40 and 50.

Datasets	*L*	2D-TP	2D-ProbS	2D-Markov	2D-NBIt	3D-IUAI	3D-IAUI	3D-IUI-IAI	MD-MP-TGPS1	MD-MP-TGPS2
MovieLens	10	0.24348	0.89773	0.25797	0.9011	0.92265	0.89989	0.96132	0.93099	0.97298
	20	0.21767	0.84351	0.23088	0.85665	0.90256	0.83715	0.93714	0.88782	0.94586
	Avg	0.20881	0.82116	0.22331	0.83641	0.8761	0.80628	0.91957	0.86343	**0.92646**
Last.fm	10	0.2328	0.87465	0.28458	0.86897	0.89766	0.69274	0.90059	0.80632	0.90004
	20	0.2078	0.84874	0.25922	0.84286	0.87055	0.65804	0.87205	0.76756	0.86906
	Avg	0.19717	0.83702	0.24891	0.83231	0.85708	0.64756	**0.86012**	0.75866	0.85635

Table 9
Intra-list diversity comparison among 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, 3D-IUI-IAI, MD-MP-TGPS1, and MD-MP-TGPS2. Column *L* refers to the length of recommendation list. Avg calculates the average value of experimental results with *L* set at 10, 20, 30, 40 and 50.

Datasets	*L*	2D-TP	2D-ProbS	2D-Markov	2D-NBIt	3D-IUAI	3D-IAUI	3D-IUI-IAI	MD-MP-TGPS1	MD-MP-TGPS2
MovieLens	10	0.845677	0.896019	0.846163	0.913051	0.935171	0.913663	0.919828	0.93154	0.939115
	20	0.86122	0.901599	0.861422	0.917339	0.934527	0.917674	0.923049	0.932043	0.941213
	Avg	0.864882	0.904097	0.865434	0.919859	0.934481	0.918957	0.924368	0.933011	**0.941356**
Last.fm	10	0.859033	0.872288	0.859147	0.874465	0.916943	0.878415	0.892066	0.875152	0.894944
	20	0.86777	0.88856	0.867815	0.891842	0.927274	0.898433	0.903969	0.892324	0.909606
	Avg	0.873399	0.894646	0.873531	0.897804	**0.930241**	0.903873	0.908562	0.897923	0.914416

Table 10
Popularity comparison among 2D-TP, 2D-ProbS, 2D-Markov, 2D-NBIt, 3D-IUAI, 3D-IAUI, 3D-IUI-IAI, MD-MP-TGPS1, and MD-MP-TGPS2. Column *L* refers to the length of recommendation list. Avg calculates the average value of experimental results with *L* set at 10, 20, 30, 40 and 50.

Datasets	*L*	2D-TP	2D-ProbS	2D-Markov	2D-NBIt	3D-IUAI	3D-IAUI	3D-IUI-IAI	MD-MP-TGPS1	MD-MP-TGPS2
MovieLens	10	117.08	68.58	116.16	55.74	45.05	59.79	50.39	46.28	35.92
	20	105.35	66.09	104.52	54.03	46.52	60.55	49.47	46.78	36.91
	Avg	98.81	63.44	98.03	52.55	46.47	59.07	48.21	46.37	**37.03**
Last.fm	10	541.37	301.85	525.13	301.55	240.23	389.93	273.93	342.95	262.16
	20	441.33	254.53	428.19	254.2	213.91	326.01	237.83	291.54	230.39
	Avg	407.95	238.71	395.82	237.86	**201.98**	302.3	223.19	271.48	216.96

some typical lengths. As is depicted, the presented MD-MP-TGPS2 method provides lower popularity, compared with the other methods, signifying better novelty/surprise of recommendations.

The strength of the proposed method MD-MP-TGPS2 is overwhelming when compared with other methods in MovieLens. The average popularity presents an obvious reduction, by 53.07% with 2D-TP, by 71.32% with 2D-ProbS, 164.73% with 2D-Markov, 41.91% with 2D-NBIt, 25.49% with 3D-IUAI, 59.52% with 3D-IAUI, and 30.19% with 3D-IUI-IAI. In Last.fm, the average popularity shows a similar reduction, by 33.45% with 2D-TP, by 10.02% with 2D-ProbS, 82.44% with 2D-Markov, 9.63% with 2D-NBIt,

Algorithm 1
MD-MP-TGPS1 (The vote-accumulation mechanism)

Input: Given the target user U_i; the length of recommendation list L; the list set of top L recommended items and their scores $\mathscr{L}_{\mathscr{P}}^{U_i}${item}[resource]via multi-dimensional meta-paths temporal graph probabilistic spreading according to Eqs. (12-14), (17-19).

Output: A list of top L recommended items $\mathscr{L}^{U_i}$for the target user U_i.

$\mathscr{L}^{U_i}${item}[vote_number, resource]=∅
foreach meth-path $p \in \mathscr{P}$ **do**
  foreach item in $\mathscr{L}_p^{U_i}$ **do**
    if item in $\mathscr{L}^{Ui}$
      $\mathscr{L}^{Ui}$[item][0]+=1
      $\mathscr{L}^{Ui}$[item][1]+=$\mathscr{L}_p^{U_i}$[item]
    else
      $\mathscr{L}^{Ui}$.Add(item,[0,0])
      $\mathscr{L}^{Ui}$[item][0]+=1
      $\mathscr{L}^{Ui}$[item][1]+=$\mathscr{L}_p^{U_i}$[item]
  end for
end for
sort($\mathscr{L}^{Ui}$, key=($\mathscr{L}^{Ui}$[item][0] Desc, $\mathscr{L}^{Ui}$[item][1] Desc), L)
return $\mathscr{L}^{Ui}$

Algorithm 2
MD-MP-TGPS2 (The maximum-rearrangement mechanism)

Input: Given the target user U_i; the length of recommendation list L; the list set of top L recommended items and their scores $\mathscr{L}_{\mathscr{P}}^{U_i}${item}[resource]via multi-dimensional meta-paths temporal graph probabilistic spreading according to (12-14), (17-19).

Output: A list of top L recommended items $\mathscr{L}^{Ui}$ for the target user U_i.

$\mathscr{L}^{U_i}${item}[resource]=∅
foreach meth-path $p \in \mathscr{P}$ **do**
  foreach item in $\mathscr{L}_p^{U_i}$ **do**
    if item in $\mathscr{L}^{Ui}$
      if $\mathscr{L}^{Ui}$[item]<$\mathscr{L}_p^{U_i}$[item]
        $\mathscr{L}^{Ui}$[item]=$\mathscr{L}_p^{U_i}$[item]
    else
      $\mathscr{L}^{Ui}$.Add(item,[0])
      $\mathscr{L}^{Ui}$[item]=$\mathscr{L}_p^{U_i}$[item]
  end for
end for
sort($\mathscr{L}^{Ui}$, key=($\mathscr{L}^{Ui}$item[0] Desc), L)
return $\mathscr{L}^{Ui}$

39.33% with 3D-IAUI, and 2.87% with 3D-IUI-IAI, to be exact. Amid all those compared recommendation methods, 3D-IUAI displays excellent novelty.

5.5. Tests of statistical significance

Tests of statistical significance are also performed to analyze all the experimental results in depth. We first use the Friedman nonparametric test to test whether the results generated by the ten compared algorithms differ significantly at a confidence level of 95%. For the precision indicator, we propose the following null hypothesis and alternative hypothesis. H0: there is no significant difference among the ten compared algorithms in the precision indicator, H1: there is significant difference among the ten compared algorithms in the precision indicator. The statistical results show that both p-values in MovieLens and Last.fm datasets are less than 0.001. Because these p-values are much less than the significance level of α=0.05, the null hypothesis is rejected, indicating that there is a significant difference in the ten compared algorithms in the metric of precision in both datasets. Similarly, we test the significant differences among those ten compared algorithms in other five evaluation metrics, namely recall, F1-score, hamming distance, intra-list diversity and popularity. The statistical results show that all p-values in MovieLens and Last.fm datasets in these five metrics are less than 0.001. In conclusion, the results of the Friedman nonparametric test indicate that there is a significant difference in those ten compared algorithms in all evaluation metrics in both datasets.

To further verify whether our proposed MD-MP-TGPS method is significantly different from other compared baselines, we leverage the paired-samples T test to test all paired-samples at a confidence level of 95%. For the indicator precision, the statistical results show

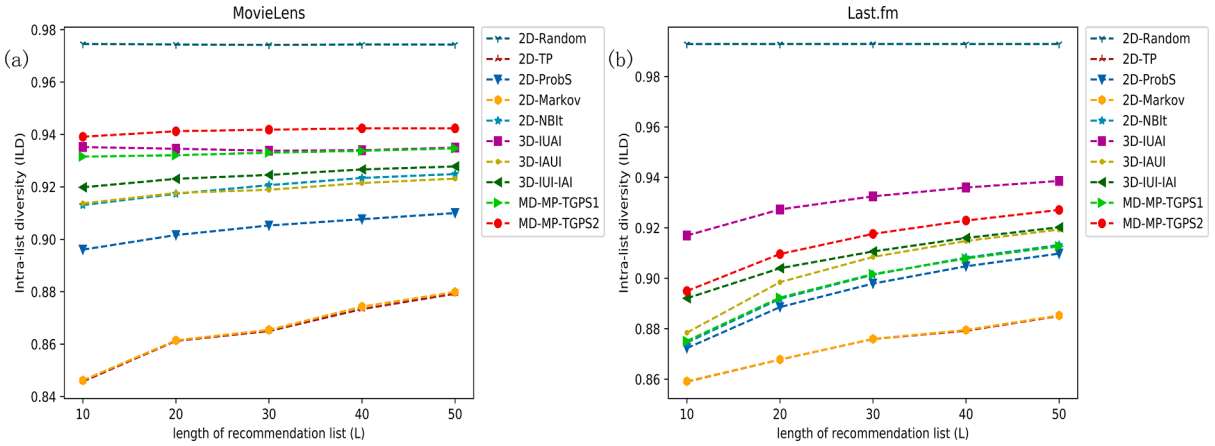


Fig. 14. Performance comparison of MD-MP-TGPS with other diversified baselines in varied recommendation lengths in intra-list diversity in (a) MovieLens and (b) Last.fm.

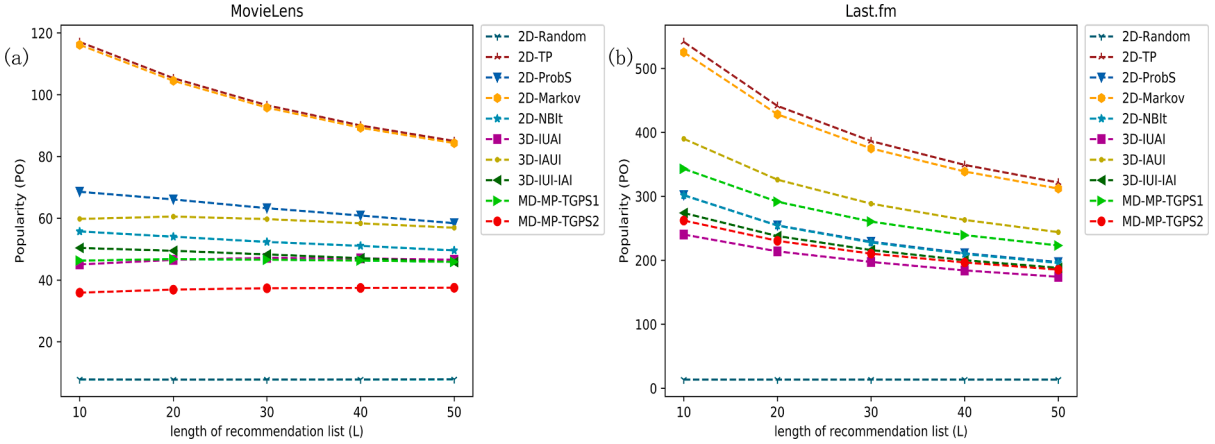


Fig. 15. Performance comparison of MD-MP-TGPS with other diversified baselines in varied recommendation lengths in popularity in (a) MovieLens and (b) Last.fm.

that there are significant differences in all paired samples, except the 2D-NBIt/MD-MP-TGPS1 pair in MovieLens dataset (Sig.=0.127) and the 3D-IUI-IAI/MD-MP-TGPS1 pair in Last.fm dataset (Sig.=0.081). For the indicator recall, there are significant differences in all paired samples, except the 2D-NBIt/MD-MP-TGPS1 pair in MovieLens dataset (Sig.=0.078) and the 3D-IUI-IAI/MD-MP-TGPS1 pair in Last.fm dataset (Sig.=0.057). For F1-score, there are significant differences in all paired samples, except the 2D-NBIt/MD-MP-TGPS1 pair in MovieLens dataset (Sig.=0.087) and the 3D-IUI-IAI/MD-MP-TGPS1 pair in Last.fm dataset (Sig.=0.064). For hamming distance, there are significant differences in all paired samples, except the 3D-IUAI/MD-MP-TGPS2 pair in Last.fm dataset (Sig.=0.41). As to intra-list diversity and popularity, significant differences are confirmed in all paired samples.

6. Conclusion and discussion

As known to all, users' preferences, items' features, and interaction behaviors between users and items are diverse and personalized. Strikingly, they all continually evolve with time. It's true that each meta-path in heterogeneous graphs represents a type of special semantic meaning. To provide a solution to desirable and personalized recommendation for users, one attractive and feasible means is to utilize multi-dimensional meta-paths while preserving rich semantic meaning simultaneously. Experimental results and statistical tests confirm that our proposed MD-MP-TGPS method achieves excellent comprehensive performance than the compared methods in aspects of accuracy, diversity and novelty of recommendation by adopting the following performance indicators of precision, recall, F1-score, hamming distance, intra-list diversity, and popularity. To be exact, taking MovieLens dataset as an example, the average precision of the proposed method is raised by 1001.51% compared with that of 2D-Random, 61.01% with 2D-TP, 47.16% with 2D-ProbS, 60.45% with 2D-Markov, 16.30% with 2D-NBIt, 86.05% with 3D-IUAI, 37.84% with 3D-IAUI, and 51.87% with 3D-IUI-IAI. The average recall shows an increase, too, namely by 892.17% with 2D-Random, 56.44% with 2D-TP, 41.30% with 2D-ProbS, 56.09% with 2D-Markov, 12% with 2D-NBIt, 72.27% with 3D-IUAI, 41.85% with 3D-IAUI, and 51.25% with D_IUI_IAI. The average

F1-score proves the superiority of our suggested method over others with an increase of 963.13% compared with that of 2D-Random, 59.21% with 2D-TP, 44.36% with 2D-ProbS, 58.65% with 2D-Markov, 14.08% with 2D-NBIt, 80.15% with 3D-IUAI, 39.3% with 3D-IAUI, and 51.06% with 3D-IUI-IAI, to be accurate. The average hamming distance is raised by 313.50% compared with that of 2D-TP, 12.82% with 2D-ProbS, 314.88% with 2D-Markov, 10.77% with 2D-NBIt, 5.75% with 3D-IUAI, 14.91% with 3D-IAUI, and 0.75% with 3D-IUI-IAI. The average intra-list diversity increases slightly, namely by 7.88% with 2D-TP, 4.12% with 2D-ProbS, 8.77% with 2D-Markov, 2.34% with 2D-NBIt, 0.74% with 3D-IUAI, 2.44% with 3D-IAUI, and 1.84% with 3D-IUI-IAI. A reduction of the average popularity is seen in the comparison between the presented method and those tested methods, 53.07% with 2D-TP, 71.32% with 2D-ProbS, 164.73% with 2D-Markov, 41.91% with 2D-NBIt, 25.49% with 3D-IUAI, 59.52% with 3D-IAUI, and 30.19% with 3D-IUI-IAI.

With regard to MD-MP-TGPS1 and MD-MP-TGPS2, MD-MP-TGPS1 has obvious advantage in accuracy metrics (higher precision, recall, and F1-score) while MD-MP-TGPS2 has a comprehensive performance advantage, showing not only good accuracy, but also excellent diversity (higher hamming distance and intra-list diversity) and novelty (lower popularity). It indicates that it is valuable and useful to apply the vote-accumulation mechanism in MD-MP-TGPS1 for improving recommendation accuracy. With the consideration of diverse recommendation results provided by temporal graph probabilistic spreading via various math-paths, MD-MP-TGPS2 comprehensively rearranges the recommended items and achieves a balance among accuracy, diversity, and novelty. As for those compared algorithms, 2D-Random has the worst accuracy and the best diversity and novelty, which can be foreseen and understood. In contrast to 2-dimentional meta-path-based ones (2D-ProbS, 2D-NBIt), 3-dimentional meta-path-based ones (3D-IUAI, 3D-IAUI, 3D-IUI-IAI) show better diversity and novelty, but lower accuracy. It is worth mentioning that the hybrid method 3D-IUI-IAI has a balanced performance in various indicators by incorporating two 2-dimensional meta-paths. It further demonstrates the feasibility and necessity of the fusion of multiple meta-paths.

Besides good accuracy, diversity and novelty, the proposed MD-MP-TGPS method also gives a potential implementation means for cross-domain recommendations which play a significant role in solving the problems of data sparsity and cold-start. For example, we first construct temporal multi-dimensional graphs for different source domains (such as movies or books). We then either aggregate these cross-domain graphs to generate a common multi-domain graph or link these cross-domain graphs by using the feature alignment of cross-domain multi-dimensional meta-paths. Finally, we adopt MD-MP-TGPS to make recommendations for each target domain separately based on the common multi-domain graph or the transfer of cross-domain knowledge. As interesting and challenging topics, such key issues as the link and aggregation of cross-domain temporal multi-dimensional graphs, the feature alignment of cross-domain multi-dimensional meta-paths, the cross-domain knowledge transfer, and cross-domain multi-dimensional meta-paths temporal graph spreading recommendations will be explored with follow-up research in the future.

Moreover, although the proposed MD-MP-TGPS method can be expanded to discretionary dimensional recommendation theoretically, we still expect trouble of addressing an excess of meta-paths in a high dimensional scenario. How to trim all the possible meta-paths and select the suitable ones is another interesting challenge to be worked out in the future.

Author Statement

No conflict of interest exits in the submission of this manuscript, and manuscript is approved by all authors for publication. I would like to declare on behalf of my co-authors that the work described was original research that has not been published previously, and not under consideration for publication elsewhere, in whole or in part. All the authors listed have approved the manuscript that is enclosed.

Acknowledgements

We acknowledge the Information Retrieval group at Universidad Autónoma de Madrid for providing us with the hetrec2011-movielens-2k and hetrec2011-lastfm-2k datasets. At the same time, we sincerely thank the editors and the reviewers for their time as well as efforts.

References

Belém, F. M., Heringer, A. G., Almeida, J. M., & Gonçalves, M. A. (2019). Exploiting syntactic and neighbourhood attributes to address cold start in tag recommendation. *Information Processing and Management, 56*(3), 771–790. https://doi.org/10.1016/j.ipm.2018.12.009

Cami, B. R., Hassanpour, H., & Mashayekhi, H. (2019). Knowledge-based systems user preferences modeling using dirichlet process mixture model for a content-based recommender system. *Knowledge-Based Systems, 163*, 644–655. https://doi.org/10.1016/j.knosys.2018.09.028

Campana, M. G., & Delmastro, F. (2017). Recommender systems for online and mobile social networks: A survey. *Online Social Networks and Media, 3–4*, 75–97. https://doi.org/10.1016/j.osnem.2017.10.005

Campos, P. G., Díez, F., & Cantador, I. (2014). Time-aware recommender systems: A comprehensive survey and analysis of existing evaluation protocols. *User Modeling and User-Adapted Interaction, 24*(1–2), 67–119. https://doi.org/10.1007/s11257-012-9136-x

Castells, P., Hurley, N. J., & Vargas, S. (2015). Novelty and diversity in recommender systems. In F. Ricci, L. Rokach, & B. Shapira (Eds.), *Recommender Systems Handbook*. Boston, MA: Springer. https://doi.org/10.1007/978-1-4899-7637-6_26.

Dong, Y., Chawla, N. V., & Swami, A. (2017). Metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Part F1296* (pp. 135–144). https://doi.org/10.1145/3097983.3098036

Fu, T. Y., Lee, W. C., & Lei, Z. (2017). HIN2Vec: Explore meta-paths in heterogeneous information networks for representation learning. In *Proceedings of International Conference on Information and Knowledge Management, Part F1318* (pp. 1797–1806). https://doi.org/10.1145/3132847.3132953

Hosseini, S. A., Khodadadi, A., Alizadeh, K., Arabzadeh, A., Farajtabar, M., Zha, H., & Rabiee, H. R. (2020). Recurrent Poisson Factorization for Temporal Recommendation. *IEEE Transactions on Knowledge and Data Engineering, 32*(1), 121–134. https://doi.org/10.1109/TKDE.2018.2879796

Hu, B., Shi, C., Zhao, W. X., & Yu, P. S. (2018). Leveraging meta-path based context for top-n recommendation with a neural co-attention model. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1*, 1531–1540. https://doi.org/10.1145/3219819.3219965

Hu, Q., Han, Z., Lin, X., Huang, Q., & Zhang, X. (2019). Learning peer recommendation using attention-driven CNN with interaction tripartite graph. *Information Sciences, 479*, 231–249. https://doi.org/10.1016/j.ins.2018.12.003

Huang, H., Shen, H., & Meng, Z. (2019). Item diversified recommendation based on influence diffusion. *Information Processing and Management, 56*(3), 939–954. https://doi.org/10.1016/j.ipm.2019.01.006

Huang, Q., Chen, B., Wang, J., & Mei, T. (2014). Personalized video recommendation through graph propagation. *ACM Transactions on Multimedia Computing, Communications and Applications, 10*(4). https://doi.org/10.1145/2598779

Ji, Y., Shi, C., Fang, Y., Kong, X., & Yin, M. (2020). Semi-supervised Co-Clustering on Attributed Heterogeneous Information Networks. *Information Processing and Management, 57*(6), Article 102338. https://doi.org/10.1016/j.ipm.2020.102338

Johnson, J., & Ng, Y. K. (2017). Enhancing long tail item recommendations using tripartite graphs and markov process. In *Proceedings of 2017 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2017* (pp. 761–768). https://doi.org/10.1145/3106426.3106439

Koren, Y. (2010). Collaborative filtering with temporal dynamics. *Communications of the ACM, 53*(4), 89–97. https://doi.org/10.1145/1721654.1721677

Koren, Y., & Bell, R. (2015). Advances in collaborative filtering. In F Ricci, L Rokach, & B Shapira (Eds.), *Recommender Systems Handbook*. Boston, MA: Springer. https://doi.org/10.1007/978-1-4899-7637-6_3.

Li, H., Wang, Y., Lyu, Z., & Shi, J. (2020). Multi-task learning for recommendation over heterogeneous information network. *IEEE Transactions on Knowledge and Data Engineering, 4347*(c). https://doi.org/10.1109/tkde.2020.2983409, 1–1.

Li, J., Tang, Y., & Chen, J. (2017). Leveraging tagging and rating for recommendation: RMF meets weighted diffusion on tripartite graphs. *Physica A: Statistical Mechanics and Its Applications, 483*, 398–411. https://doi.org/10.1016/j.physa.2017.04.121

Lü, L., Medo, M., Yeung, C. H., Zhang, Y. C., Zhang, Z. K., & Zhou, T. (2012). Recommender systems. *Physics Reports, 519*(1), 1–49. https://doi.org/10.1016/j.physrep.2012.02.006

Mitrović, S., & De Weerdt, J. (2020). Churn modeling with probabilistic meta paths-based representation learning. *Information Processing and Management, 57*(2), Article 102052. https://doi.org/10.1016/j.ipm.2019.06.001

Molaei, S., Zare, H., & Veisi, H. (2020). Deep learning approach on information diffusion in heterogeneous networks. *Knowledge-Based Systems, 189*, Article 105153. https://doi.org/10.1016/j.knosys.2019.105153

Najafabadi, M. K., Mohamed, A., & Onn, C. W. (2019). An impact of time and item influencer in collaborative filtering recommendations using graph-based model. *Information Processing and Management, 56*(3), 526–540. https://doi.org/10.1016/j.ipm.2018.12.007

Poulain, R., & Tarissan, F. (2020). Investigating the lack of diversity in user behavior: The case of musical content on online platforms. *Information Processing and Management, 57*(2), Article 102169. https://doi.org/10.1016/j.ipm.2019.102169

Sánchez, P., & Bellogín, A. (2019). Building user profiles based on sequences for content and collaborative filtering. *Information Processing and Management, 56*(1), 192–211. https://doi.org/10.1016/j.ipm.2018.10.003

Sánchez, P., & Bellogín, A. (2020). Time and sequence awareness in similarity metrics for recommendation. *Information Processing and Management, 57*(3), Article 102228. https://doi.org/10.1016/j.ipm.2020.102228

Shi, C., Han, X., Song, L., Wang, X., Wang, S., Du, J., & Yu, P. S. (2021). Deep collaborative filtering with multi-aspect information in heterogeneous networks. *IEEE Transactions on Knowledge and Data Engineering, 33*(4), 1413–1425. https://doi.org/10.1109/TKDE.2019.2941938

Shi, C., Hu, B., Zhao, W. X., & Yu, P. S. (2019). Heterogeneous information network embedding for recommendation. *IEEE Transactions on Knowledge and Data Engineering, 31*(2), 357–370. https://doi.org/10.1109/TKDE.2018.2833443

Shi, C., Kong, X., Huang, Y., Yu, P. S., & Wu, B. (2014). HeteSim: A general framework for relevance measure in heterogeneous networks. *IEEE Transactions on Knowledge and Data Engineering, 26*(10), 2479–2492. https://doi.org/10.1109/TKDE.2013.2297920

Shi, C., Zhang, Z., Luo, P., Yu, P. S., Yue, Y., & Wu, B. (2015). Semantic Path based Personalized Recommendation on Weighted Heterogeneous Information Networks. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management* (pp. 453–462). https://doi.org/10.1145/2806416.2806528

Sun, Y., Han, J., Yan, X., Yu, P. S., & Wu, T. (2011). Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of the VLDB Endowment, 4*(11), 992–1003. https://doi.org/10.14778/3402707.3402736

Wang, X., Lu, Y., Shi, C., Wang, R., Cui, P., & Mou, S. (2020). Dynamic heterogeneous information network embedding with meta-path based proximity. *IEEE Transactions on Knowledge and Data Engineering, 57*(2). https://doi.org/10.1109/TKDE.2020.2993870, 1–1.

Wang, Y., & Han, L. (2020). Personalized recommendation via network-based inference with time. *Physica A: Statistical Mechanics and Its Applications, 550*, Article 123917. https://doi.org/10.1016/j.physa.2019.123917

Wang, Y., & Han, L. (2021). Adaptive time series prediction and recommendation. *Information Processing and Management, 58*(3), Article 102494. https://doi.org/10.1016/j.ipm.2021.102494

Xiang, L., Yuan, Q., Zhao, S., Chen, L., Zhang, X., Yang, Q., & Sun, J. (2010). Temporal recommendation on graphs via long- and short-term preference fusion. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 723–731). https://doi.org/10.1145/1835804.1835896

Xie, F., Zheng, A., Chen, L., & Zheng, Z. (2021). Attentive Meta-graph Embedding for item Recommendation in heterogeneous information networks. *Knowledge-Based Systems, 211*, Article 106524. https://doi.org/10.1016/j.knosys.2020.106524

Xin, M., & Wu, L. (2020). Using multi-features to partition users for friends recommendation in location based social network. *Information Processing and Management, 57*(1), Article 102125. https://doi.org/10.1016/j.ipm.2019.102125

Yu, F., Zeng, A., Gillard, S., & Medo, M. (2016). Network-based recommendation algorithms: A review. *Physica A: Statistical Mechanics and Its Applications, 452*, 192–208. https://doi.org/10.1016/j.physa.2016.02.021

Zhang, X., & Chen, L. (2020). mSHINE: A Multiple-meta-paths Simultaneous Learning Framework for Heterogeneous Information Network Embedding. *IEEE Transactions on Knowledge and Data Engineering, (Early Access)*. https://doi.org/10.1109/tkde.2020.3025464

Zhang, Z. K., Zhou, T., & Zhang, Y. C. (2010). Personalized recommendation via integrated diffusion on user–item–tag tripartite graphs. *Physica A: Statistical Mechanics and Its Applications, 389*(1), 179–186. https://doi.org/10.1016/j.physa.2009.08.036

Zhang, Z., Huang, J., Tan, Q., Sun, H., & Zhou, Y. (2021). CMG2Vec: A composite meta-graph based heterogeneous information network embedding approach. *Knowledge-Based Systems, 216*, Article 106661. https://doi.org/10.1016/j.knosys.2020.106661

Zhao, H., Yao, Q., Li, J., Song, Y., & Lee, D. L. (2017). Meta-graph based recommendation fusion over heterogeneous information networks. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Part F1296* (pp. 635–644). https://doi.org/10.1145/3097983.3098063

Zhou, T., Kuscsik, Z., Liu, J. G., Medo, M., Wakeling, J. R., & Zhang, Y. C. (2010). Solving the apparent diversity-accuracy dilemma of recommender systems. *Proceedings of the National Academy of Sciences of the United States of America, 107*(10), 4511–4515. https://doi.org/10.1073/pnas.1000488107

Zhou, W., & Han, W. (2019). Personalized recommendation via user preference matching. *Information Processing and Management, 56*(3), 955–968. https://doi.org/10.1016/j.ipm.2019.02.002