Dual Graph Neural Networks for Dynamic Users' Behavior Prediction on Social Networking Services

Junwei Li, Le Wu, Yulu Du, Richang Hong, *Senior Member, IEEE*, and Weisheng Li, *Member, IEEE*

***Abstract*—Social network services (SNSs) provide platforms where users engage in social link behavior (e.g., predicting social relationships) and consumption behavior. Recent advancements in deep learning for recommendation and link prediction explore the symbiotic relationships between these behaviors, leveraging social influence theory and user homogeneity, i.e., users tend to accept recommendations from social friends and connect with like-minded users. These studies yield positive feedback for users and platforms, fostering practical applications and economic development. While previous works jointly model these behaviors, most studies often overlook the evolution of social relationships and users' preferences in dynamic scenes and the correlations inside, as well as the higher order information within the social network and preference network (consumption history). To address this, we propose the dynamic graph neural joint behavior prediction model (DGN-JBP). Specifically, we actively disentangle and initialize user embeddings from multiple perspectives to refine information for modeling. Additionally, we design an attentive graph neural network and combine it with gate recurrent units (GRUs) to extract high-order dynamic information. Finally, we design a dual framework and purposefully fuse embeddings to mutually enhance the effectiveness of predictions on two prediction tasks. Extensive experimental results on two real-world datasets clearly demonstrate the effectiveness of our proposed model.**

***Index Terms*—Dynamic information, graph neural networks (GNNs), social networking services (SNSs), user behaviors modeling.**

I. Introduction

WITH the rapid evolution of the Internet, social networking services (SNSs) dominated by user behaviors have undergone dynamic changes, showcasing diversity in form, content, business, data application, and interaction complexity.

Manuscript received 2 March 2024; revised 22 May 2024; accepted 26 May 2024. This work was supported in part by the Natural Science Foundation of Chongqing from Chongqing Municipal Science and Technology Bureau under Grant CSTB2023NSCQ-MSX0613, and in part by the Science and Technology Research Program of Chongqing Municipal Education Commission under Grant KJQN202200649. *(Corresponding author: Weisheng Li.)*

Junwei Li and Weisheng Li are with Chongqing Key Laboratory of Image Cognition, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: lijw@cqupt.edu.cn; liws@cqupt.edu.cn).

Le Wu and Richang Hong are with the School of Computer and Information, Hefei University of Technology, Hefei, Anhui 230009, China (e-mail: lewu.ustc@gmail.com; hongrc.hfut@gmail.com).

Yulu Du is with Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: duyl@cqupt.edu.cn).

Digital Object Identifier 10.1109/TCSS.2024.3409383

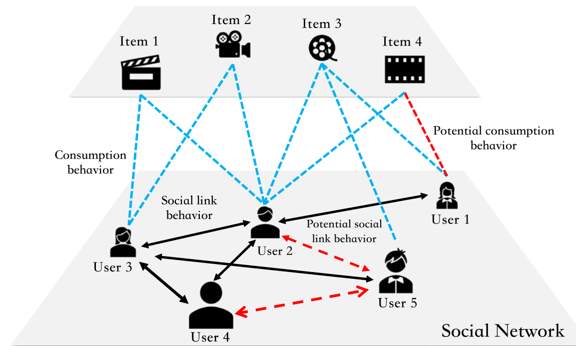

Fig. 1. Two kinds of user behaviors on SNSs.

Serving as a daily tool widely accessed by people, SNSs provide platforms for users to connect with others and share their preferences with social friends. Take popular apps such as TikTok, for example, users follow others they like, and their video preferences are recommended to social followers and friends. Beyond providing excellent services, SNSs yield substantial profits. As a unique and emerging technology, SNS-based services, including live commerce, have played a crucial role in mitigating the economic impact of the COVID-19 pandemic over the past three years. They have become a vital force for stabilizing the national economy, showcasing unique advantages in the era of big data. The wealth of information in the big data era creates fertile ground for mining user interests and social relationships on these platforms. Consequently, accurately modeling users' primary behaviors, social link behavior [1] (predicting possible social links), and consumption behavior [2] (recommending potential items of interest) have become core tasks with practical value and a research hotspot, which hold the potential to benefit various applications in the real world.

In the past, sociologists have confirmed that these two behaviors mutually promote each other, i.e., users are more likely to be recommended by social friends (e.g., user 1 considers item 4 by the recommendation of a social friend user 2), and those with similar interests easily establish social links (e.g., users 2 and 5 become friends because they have similar interests in item 3) [3], the interactive relationships between the two behaviors are shown in Fig. 1. For instance, on TikTok, individual video content verticality is crucial for gaining attention from fan links. Ignoring social relationships overlooks a key factor in online social networking—the correlation of interests among users. Also, higher acceptance of recommendations from social connections drives opportunities in live streaming, creating new entrepreneurial and employment prospects. Therefore, some studies focus on information collaboration in independent behavior prediction tasks by using one behavior as auxiliary data to enhance the accuracy of another prediction task [4], [5].

The aforementioned ideas primarily focus on predicting specific user behaviors while ignoring modeling the close correlations between them. By leveraging the inherent advantages of SNSs, the records of users' two behaviors are aggregated and can be effectively utilized. In contrast to prior methods, there are works that leverage the connectivity of user behavior and jointly model these two behaviors, making them mutually enhance each other and further improve the performance of both prediction tasks. For example, friendship and interest propagation (FIP) was proposed to jointly model two behaviors by sharing the same user embeddings [6]. With the development of machine learning, deep learning-based models have been widely studied, and researchers explored the modeling process by using deep neural network [7] and network embedding methods [8].

However, most existing methods for predicting user behavior are grounded in current behavior records and primarily rely on static modeling due to the faster rate of market iteration and update, as well as the natural social relationships and evolutionary characteristics of interest preferences of users, traditional static models are challenging to provide personalized predictions in many complex scenarios, therefore, modeling the evolution of user behavior over time is of great significance. Moreover, the accuracy of prediction outcomes hinges significantly on the quality of learned embeddings; the existing models face challenges in capturing high-order relationship information during dynamic evolution, which ultimately limits their performance. Additionally, the mutual promotion of the two behaviors needs to be designed in the modeling stage and is underrepresented in many efforts. Therefore, rethinking how to design a simple yet efficient structure that seamlessly integrates both behaviors is worthwhile.

To address these challenges, in this article, we propose the dynamic graph neural joint behavior prediction model (DGN-JBP), which could capture the dynamic natures of user behaviors, aggregate the higher order information of graphs, and model the intricate interactions between behaviors. Specifically, based on the time stamp of behavior data, we create user behavior tensors to record the data for calculation. Furthermore, we argue that the quality of user embeddings is crucial for subsequent modeling, given that users combine the social network and the interest network. Therefore, unlike other works, we actively disentangle and initialize user embeddings from different perspectives and design an attentive graph neural structure to aggregate high-order information right from the outset. Additionally, we extract temporal characteristics by combining the user behavior tensors with recurrent neural networks (RNNs). Subsequently, we proposed an elaborated dual framework to fuse relevant embeddings based on the relationships between user behaviors, enabling them to mutually enhance two prediction tasks. Finally, we present extensive experimental results on two real-world datasets, clearly demonstrating the effectiveness of our proposed model. In summary, our main contributions are listed as follows.

1) To refine information for modeling, we actively disentangle user embeddings with multiperspectives from created user behavior tensors, which has not been given much attention in previous works.
2) We propose an end-to-end model, which can be described as two steps. In the first step, we design an attention neural network to flexibly aggregate disentangled user embeddings, which are subsequently combined with gate recurrent units (GRUs) to extract high-order dynamic information. In the second step, we propose a dual framework to capture the mutual correlations between two behaviors by different combinations of embeddings, which has been rarely studied in previous works.
3) Extensive experimental results on two real-world datasets demonstrate the effectiveness and superiority of *DGN-JBP* over similar works. Additionally, we proved the correlation between two types of user behaviors and their mutual reinforcement, which will contribute to similar areas in user behavior analysis.

II. Problem Definition

For ease of presentation, we use lowercase alphabets (e.g., a) to denote scalars, boldface and italic lowercase alphabets (e.g., $\boldsymbol{a}$) to denote vectors, italic uppercase alphabets (e.g., A) to denote set and boldface uppercase alphabets (e.g., $\mathbf{A}$) to denote matrices. Throughout, we will use a, b, c to denote the users and i, j, k to denote the items.

Suppose we have n users $U = \{u_1, u_2, \ldots, u_n\}$ and m items $V = \{v_1, v_2, \ldots, v_m\}$, then a social network can be denoted as matrix $\mathbf{Y} \in \mathbb{R}^{n\times n}$. If user a trusts user b, the (a, b)th entry y_{ab} of $\mathbf{Y}$ equals 1, otherwise it equals 0. Similarly, we use $\mathbf{R} \in \mathbb{R}^{n\times m}$ to denote the rating matrix and r_{ai} represents the rating score that user a rated on item i. Besides, by using the timestamp information, we define user behavioral tensors $T_S \in \mathbb{R}^{t\times n\times n}$ and $T_R \in \mathbb{R}^{t\times n\times m}$, which can also be described as matrix sequence $T_S = \{\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_t\}$ and $T_R = \{\mathbf{R}_1, \mathbf{R}_2, \ldots, \mathbf{R}_t\}$.

Based on this, we briefly introduce the embedding initialization. First, since the superior performance of combining deep and shallow embeddings [9], we represent users and items in both deep and shallow space. The shallow embeddings model the static and simple interactions, while the deep embeddings capture complex and dynamic factors. Second, due to the diversification of user behavior intentions, setting the same user embedding will limit the final results [7]. Thus, we represent various user intentions by different user embeddings. As a bridge connecting two behavior graphs, user embeddings are divided into two spaces, i.e., social space and preference space. Third, as we propose a time-aware model for joint behavior prediction, to capture the dynamic characteristic, we further divide the deep embeddings of users and items into two parts: 1) the static deep embeddings are used to depict the static but complex interactions; and 2) the dynamic deep embeddings are used

TABLE I
MAIN NOTATIONS OF DIFFERENT LATENT SPACES

Notations	Interpretation
$\mathbf{S}$	Deep dynamic user social latent space
$\mathbf{P}$	Deep dynamic user preference latent space
$\mathbf{Q}$	Deep dynamic item latent space
$\mathbf{C}$	Deep static user social latent space
$\mathbf{D}$	Deep static user preference latent space
$\mathbf{E}$	Deep static item latent space
$\mathbf{X}$	Shallow user social latent space
$\mathbf{W}$	Shallow user preference latent space
$\mathbf{M}$	Shallow item latent space

to model the characteristics of users and items changing over time. The reason for establishing a more refined disentangled embedding space is that the previous works often ignore the importance of embedding establishment in the initial stage of model training and utilize vaguely defined embeddings, which will make the model amplify the inherent errors during the learning process. In summary, the abovementioned embeddings are listed in Table I.

Problem Definition: The input data of our prediction task include a set of users U, a set of items V, and two kinds of users' behavior records T_S and T_R. Then, the goal contains three aspects: 1) design an effective joint model to output a ranking list of potential preference items for each target user; 2) based on existing social network and neighbor preferences, predict social links that each target user may create in the future; and 3) proving the mutual promotion between two user behaviors.

III. Related Work

A. Social Link and Consumption Behavior Prediction

As a common task, link prediction has received extensive attention and research across various domains [10], such as biology, medicine, and knowledge graph-related applications. Traditional unsupervised methods often tackle the link prediction problem by converting it into node proximity based on network topology [1], [11], [12], [13]. In contrast, supervised learning-based approaches treat existing links as positive samples and design various models to calculate the probability of links that would be created between nodes [14]. With the evolution of deep learning, researchers are increasingly emphasizing the application of deep neural networks (DNNs) [10], the relevant literature can be classified into two categories: DNN-based methods and network embedding methods. DNN-based methods primarily enhance prediction accuracy through the nonlinear fitting capabilities [15], [16]. Additionally, network embedding methods utilize the network topology to capture higher order information, and the learned low-latent embeddings are friendly for the subsequent applications [17].

Recommender systems have also been well applied to consumption behavior prediction [18], [19]. As a representative technology, collaborative filtering (CF) emerges as a highly effective approach [2] by leveraging associations among like-minded users to achieve superior results. Recently, DNN-based recommender systems [20], [21], [22] have been extensively explored to compensate for the limited model fitting ability and to integrate information from multiple sources [23]. As users tend to be influenced by social friends when making consumption decisions [5], [24], the social recommendation has gained widespread attention, aiming to alleviate the sparsity issue arising from insufficient consumption records while enhancing prediction performance [25], [26], [27] and exploit the multifacet social relations for enhanced item recommendation [28].

In conclusion, the mature development of link prediction and social recommendation provides robust theoretical and experimental support for research on joint behavior prediction on SNS platforms.

B. Joint Model for Two Kinds of Behaviors

In fact, sociologists have confirmed that users' social link behavior and consumption behavior are not independent and unrelated. Instead, they coexist and interact with each other [3], [24]. To verify this mutual promotion, some studies jointly model two types of user behaviors. As a representative of these works, Yang et al. proposed a unified framework FIP, which shared the same user embeddings for predictions [6], and the experimental results indicate their mutual promotion. Building upon these findings, Wu et al. proposed evolutional latent joint prediction (ELJP) to extend the work to the temporal scenario [29]. To model the complex interactions between behaviors, researchers also apply deep learning techniques to verify the effectiveness. For example, Neural joint modeling (NJM) [30] achieved a better performance than FIP and ELJP by taking advantage of DNNs. Besides, Li et al. proposed neural joint behavior prediction (NJBP), which has a dual DNN structure for prediction, and the social link prediction loss is redefined to have the ability to capture high-order friendships [7]. Since user behavior data can constitute the graph structure, Li et al. studied the effect of the network embedding method on behavior prediction tasks [8].

C. Graph Neural Networks (GNNs) for Behavior Prediction and Other Applications

As a powerful and widely used deep learning technique, GNNs have demonstrated significant success in graph representation learning [31] in both homogeneous [32] and heterogeneous [33] graph-based researches, which aim to learn multi-hop information from neighborhoods for downstream tasks and applications [34], [35], [36], [37]. Moreover, graph convolutional networks (GCNs), as a representative model of GNNs, have attracted widespread attention and research regarding the smoothing issue after multilayer operations [38], [39].

As for the task of user behavior prediction, given that the learned final embeddings directly impact prediction accuracy, GNNs can be applied to both social link and consumption behavior prediction tasks. Typically, most of the GNN-based models design the framework from a graph perspective and generalize convolutional operations to enhance the embeddings. For instance, Zhang et al. proposed SEAL to extract a local enclosing subgraph around it and use a GNN to learn general graph structure features for link prediction. Also, Fan et al. proposed GraphRec, which considers three kinds of graphs from users' two kinds of behavior and fuses first-order social and first-order item neighbors with nonlinear neural networks [40]. To get high-order information, Wu et al. proposed GCN-based models to study the information diffusion on the social networks and the interest network [26], [41]. In efforts to explore their joint effects, Xiao et al. proposed MGNN to take advantage of GNNs to predict two kinds of behavior [42].

As social relationships and user preferences evolve over time, researchers also introduce time-aware deep learning models to capture dynamic features. Manessi et al. proposed dynamic GCN, which jointly exploits structured data and temporal information through the use of a neural network model [43]. Wu et al. presented a novel architecture for the session-based recommendation that incorporates graph models into representing session sequences [44]. Chen et al. propose a new deep learning model GC-LSTM, which combines LSTM and GCN for dynamic network link prediction [45].

Although existing models have made some improvements compared to the previous works, there are still areas for improvement. First, previous works have rarely studied the initialization methods of embeddings and their roles in joint behavior modeling. Therefore, the first aim of our work is the accurate utilization and modeling of embedding. Also, as a method for extracting high-order information, GCN has gained widespread attention and usage. However, in the joint modeling of user behavior, how to utilize actively disentangled high-order embeddings for precise modeling has not been thoroughly investigated. Finally, the above-mentioned methods rarely utilize high-order information for the dynamic joint modeling of user behavior on SNSs. The utilization of RNNs and their effectiveness still leave room for further research. Besides, our proposed model leverages only user behavior records and the corresponding time stamps for modeling. The adaptability to diverse datasets offered by this method will benefit existing research on user behavior prediction in both academia and industry.

IV. The Proposed Model

In this section, we first give a brief introduction to our proposed end-to-end model. To clearly describe the process, we divide it into two steps.

1) We actively disentangle user embeddings and aggregate high-order information from the neighborhood for extracting the following dynamic characteristics, as shown in Fig. 2.
2) We elaborately design a dual framework and make full use of the learned high-order dynamic embeddings for meaningful combinations to predict accurate results, as shown in Fig. 3.

A. High-Order Dynamic Embeddings Modeling

First, we initialize original embeddings from two networks, the user social embeddings from the social network are employed for social link prediction. Similarly, in the interest network, we have user preference embeddings and item embeddings for consumption behavior prediction. This approach allows us to assign distinct roles to these embeddings (e.g., user social embedding directly influenced by social link relationships). Subsequently, for each embedding mentioned above, we further divided them into three types: deep static embeddings, deep dynamic embeddings, and shallow embeddings. As outlined in Problem Definition, these nine embeddings could handle most of the situations based on the previous researches [7], [9], and due to the evolving nature of user social relationships and preferences, the deep dynamic embeddings are very important for achieving optimal results.

1) High-Order Embeddings Generation: To get the deep dynamic embeddings with high-order information, we first adopt GATs to purposefully aggregate neighbors' information for different types of initialized deep dynamic embeddings, i.e., deep dynamic user social embeddings, deep dynamic user preference embeddings, and deep dynamic item embeddings. We denote $Y_a = [b|y_{ab} = 1]$ and $R_a = [i|r_{ai} = 1]$, similarly, the user set that consumed the item i is represented as $R_i = [a|r_{ia} = 1]$. The update methods are listed as follows:

$$\tilde{\boldsymbol{s}}^l_{a,t}, = \mathrm{AGG}_{(u,u)}(\boldsymbol{s}^{l-1}_{b,t}, \forall b \in Y_a) = \sum_{b\in Y_a} \alpha_{ba}\boldsymbol{s}^{l-1}_{b,t}$$
$$\boldsymbol{s}^l_{a,t} = \tilde{\boldsymbol{s}}^l_{a,t} + \boldsymbol{s}^{l-1}_{a,t} \tag{1}$$
$$\tilde{\boldsymbol{p}}^l_{a,t}, = \mathrm{AGG}_{(u,v)}(\boldsymbol{q}^{l-1}_{j,t}, \forall j \in R_a) = \sum_{j\in R_a} \beta_{ja}\boldsymbol{q}^{l-1}_{j,t}$$
$$\boldsymbol{p}^l_{a,t} = \tilde{\boldsymbol{p}}^{l-1}_{a,t} + \boldsymbol{p}^{l-1}_{a,t} \tag{2}$$
$$\tilde{\boldsymbol{q}}^l_{i,t}, = \mathrm{AGG}_{(u,v)}(\boldsymbol{p}^{l-1}_{b,t}, \forall b \in R_i) = \sum_{b\in R_i} \beta_{ib}\boldsymbol{p}^{l-1}_{b,t}$$
$$\boldsymbol{q}^l_{i,t} = \tilde{\boldsymbol{q}}^l_{i,t} + \boldsymbol{q}^{l-1}_{i,t} \tag{3}$$

where $\boldsymbol{s}^l_{a,t}$ and $\boldsymbol{p}^l_{a,t}$ are the deep dynamic social embeddings and deep dynamic preference embeddings of user a at stamp t and layer l, and $\boldsymbol{q}^l_{i,t}$ is the deep dynamic item embedding of item i at t and layer l, α_{ba}, β_{ja} and β_{ib} are the attention weight when aggregating the lth order neighbors' embeddings.

2) Dynamic Attributes Modeling: Now we have three types of high-order embeddings of lth layer without dynamic information, since social relationships, user preferences, and item attributes change over time, consequently, relative to modeling static characteristics, we model these dynamic attributes by incorporating GRUs. Actually, the embeddings extracted by GATs in the previous part are well suited for the two kinds of tasks, since each stamp corresponds to an individual graph, which could automatically aggregate topological information with considerations of rich node connections. Consequently, the user embeddings of each time stamp after lth convolutional layer could be considered as the input of the GRUs, we present the update functions of deep dynamic user social embeddings as follows:

$$\boldsymbol{z}^l_{s,t} = \sigma(\mathbf{W}^s_z[\boldsymbol{h}_{s,t-1}, \mathbf{S}^l_t] + \boldsymbol{b}^s_z) \tag{4}$$
$$\boldsymbol{r}^l_{s,t} = \sigma(\mathbf{W}^s_r[\boldsymbol{h}_{s,t-1}, \mathbf{S}^l_t] + \boldsymbol{b}^s_r) \tag{5}$$
$$\tilde{\boldsymbol{h}}^l_{s,t} = \tanh(\mathbf{W}^s_h[\boldsymbol{r}^l_{s,t} * \boldsymbol{h}_{s,t-1}, \mathbf{S}^l_t] + \boldsymbol{b}^s_h) \tag{6}$$
$$\boldsymbol{h}^l_{s,t} = (1 - \boldsymbol{z}^l_{s,t}) * \boldsymbol{h}^l_{s,t-1} + \boldsymbol{z}^l_{s,t} * \tilde{\boldsymbol{h}}^l_{s,t} \tag{7}$$

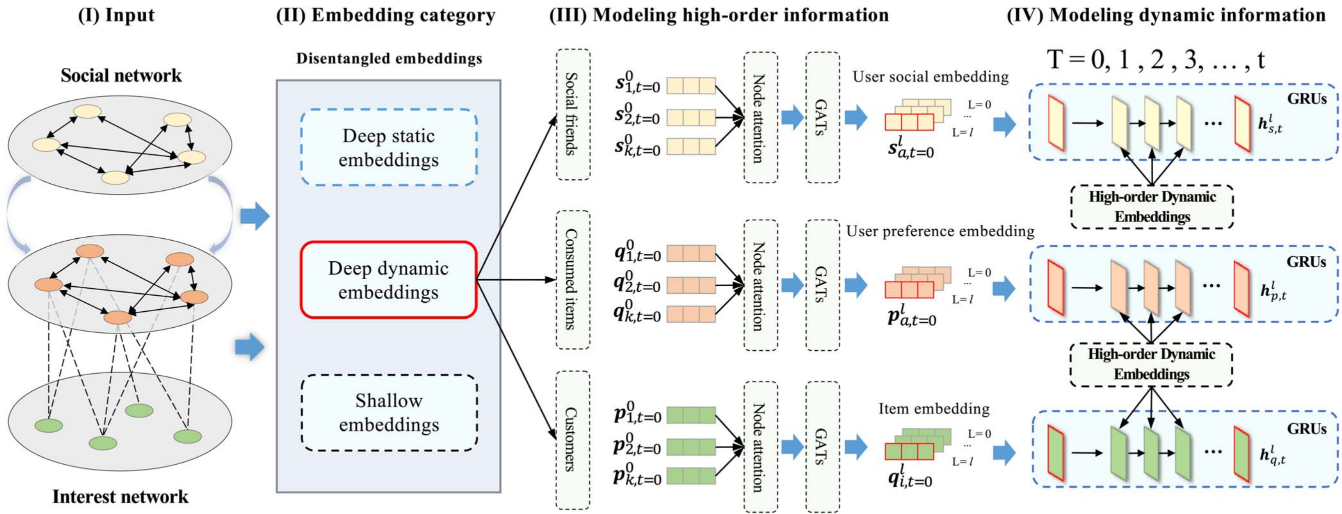

Fig. 2. Overall process of disentangled embeddings initialization and high-order dynamic embeddings generation. We utilize graph attention networks (GATs) to aggregate high-order information from disentangled embeddings from graphs and capture dynamic information by GRUs.

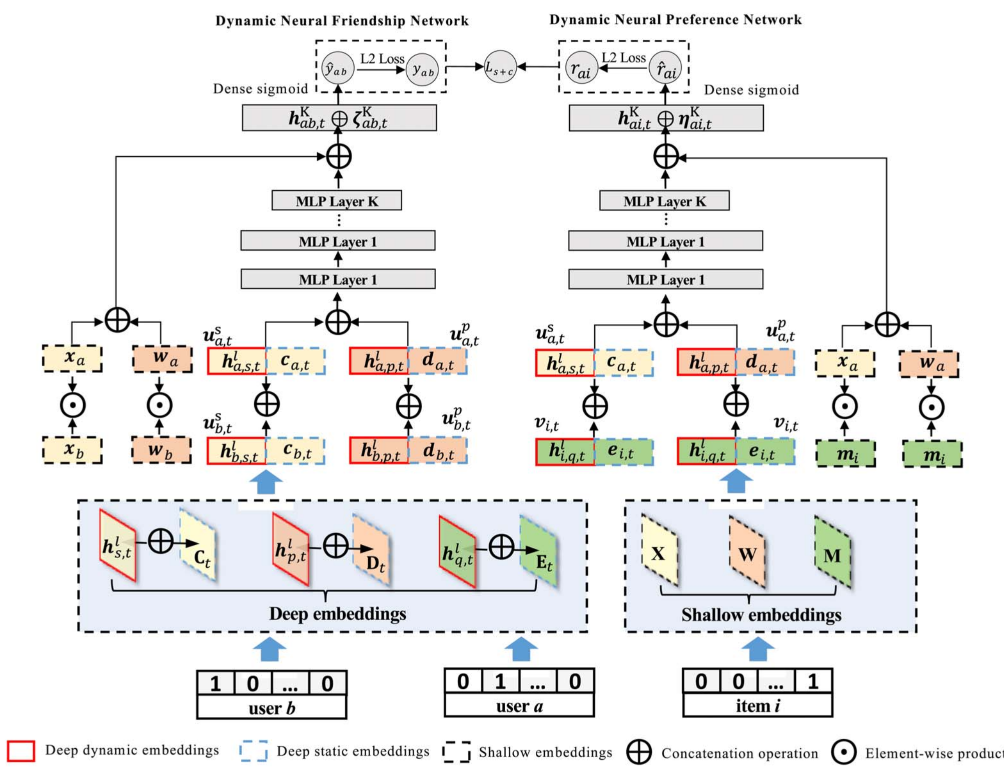

Fig. 3. Dual framework for predictions. Specifically, the dynamic neural friendship network (DNFN) on the left aims at modeling users' social link behavior, and the dynamic neural preference network (DNPN) on the right focuses on consumption behavior. The mutual influence and interaction between two behaviors are achieved through embedding fusion.

where $\boldsymbol{z}_{s,t}^{l}$ and $\boldsymbol{r}_{s,t}^{l}$ are the update and reset gates for $\mathbf{S}$ at t, respectively. For each t, the hidden state $\boldsymbol{h}_{s,t}^{l}$ remains the high-order dynamic information of social relationships. Similarly, we also get high-order dynamic user preference embeddings $\boldsymbol{h}_{p,t}^{l}$ and dynamic item embeddings $\boldsymbol{h}_{q,t}^{l}$ in the same way.

B. DGN-JBP Model

As illustrated in Fig. 3, the dual framework contains two components: the DNFN on the left aims at modeling social link behavior, while the DNPN on the right focuses on consumption behavior. Additionally, we emphasize that the high-order dynamic embeddings are extracted in the preceding stage. Since it is an end-to-end model, the dynamic characteristics will be captured during training.

1) DNFN Modeling:

a) Input layer: As each user and item have a one-hot encoding, the input layer reads a batch of one-hot encodings from behavior records, preparing for selecting the corresponding embeddings to perform the next layer.

b) Embedding layer: The embedding layer receives the one-hot encodings from the input layer and extracts relevant embeddings from the corresponding latent spaces through a lookup operation. Notably, for all deep embeddings, we concatenate high-order dynamic and static embeddings to enhance the generalizability

$$\mathbf{U}_t^s = \boldsymbol{h}_{s,t}^l \oplus \mathbf{C}_t \tag{8}$$

$$\mathbf{U}_t^p = \boldsymbol{h}_{p,t}^l \oplus \mathbf{D}_t \tag{9}$$

$$\mathbf{V}_t = \boldsymbol{h}_{q,t}^l \oplus \mathbf{E}_t \tag{10}$$

we omit the superscript l in the following content since we utilize the embeddings learned from the final layer l of GATs.

c) Fusion layer: The fusion layer simulates the interactions of social link behavior. On one hand, we adopt the elementwise product operation for shallow embeddings to preserve the static and simple interactive information, i.e., $\boldsymbol{x}_a \odot \boldsymbol{x}_b$ and $\boldsymbol{w}_a \odot \boldsymbol{w}_b$, the former describes the establishment of social relationships through some of the static and simple factors in the social networks, while the latter indicates that users establish social relationships based on their own similar preferences. To provide shallow embeddings for subsequent training, we concatenate the raw shallow elementwise embeddings as

$$\begin{aligned}\boldsymbol{\zeta}_{ab}^s &= \mathrm{CONCAT}(\boldsymbol{x}_a \odot \boldsymbol{x}_b, \boldsymbol{w}_a \odot \boldsymbol{w}_b)\\ &= \boldsymbol{x}_a \odot \boldsymbol{x}_b \oplus \boldsymbol{w}_a \odot \boldsymbol{w}_b.\end{aligned} \tag{11}$$

On the other hand, since the interactions of deep embeddings encapsulate the dynamics and complex uncertainties of user relationships, we leverage deep neural networks for information extraction on the basis of dynamic and static information. Consequently, we concatenate all the deep embeddings that are sent by the previous layer

$$\boldsymbol{\eta}_{ab,t}^s = \boldsymbol{u}_{a,t}^s \oplus \boldsymbol{u}_{b,t}^s \oplus \boldsymbol{u}_{a,t}^p \oplus \boldsymbol{u}_{b,t}^p. \tag{12}$$

We input the concatenated embedding $\boldsymbol{\eta}_{ab,t}^s$ into a forward deep neural network for training. For layer $k = 1, \ldots, K$, the number of neurons is halved in each layer, resulting in the output of the Kth layer

$$\boldsymbol{h}_{ab,t}^K = \mathrm{DNN}_{K-1}^s(\ldots\mathrm{DNN}_1^s(\mathbf{Z_1}^T \cdot \boldsymbol{\eta}_{ab,t}^s) + \boldsymbol{b}_1\ldots) + \boldsymbol{b}_{K-1} \tag{13}$$

where $\boldsymbol{Z}_K$ and $\boldsymbol{b}_K$ denote the DNNs' trainable weight matrix and bias of Kth layer.

d) Output layer: After we acquire the final deep high-level and shallow interactive embeddings, we concatenate them and employ a single layer to predict the score as

$$\hat{y}_{ab} = \sigma(\mathbf{Z}_K^T[\boldsymbol{h}_{ab,t}^K, \boldsymbol{\zeta}_{ab}^s] + \boldsymbol{b}_K). \tag{14}$$

e) Loss function: We adopted the pairwise logloss, which has been widely used for recommendation models with good performance [46], [47], then we have the following:

$$L_s = \sum_{(a,b,y_{ab},t)\in \mathbf{Y}_t^+ \cup \mathbf{Y}_t^-} (y_{ab}\log(\hat{y}_{ab}) + (1-y_{ab})\log(1-\hat{y}_{ab})) \tag{15}$$

where $\mathbf{Y}_t^+$ denotes the set of positive samples (observed user–user pairs at time stamp t), and $\mathbf{Y}_t^-$ denotes the set of negative samples (randomly sampled from $\mathbf{Y}_t$).

2) DNPN Modeling:

a) Input and embedding layer: Similarly, the input layer reads a batch of one-hot user–item pairs for the embedding layer to select corresponding embeddings.

b) Fusion layer: Since users' consumption behavior mainly comes from the recommendations of social friends (social influence) and inherent personalized preferences, we combine different user embeddings and item embeddings to model this complex process. Specifically, the combination of shallow/deep user social embeddings and item embeddings contributes to social influence modeling. Simultaneously, the shallow/deep user preferences embeddings and item embeddings are concatenated for consumption behavior prediction (inherent preferences). In this way, we simulated two response modes of consumption behavior. Consequently, for shallow embeddings, we model the simple and static interactions as follows:

$$\begin{aligned}\boldsymbol{\zeta}_{ai}^c &= \mathrm{CONCAT}(\boldsymbol{x}_a \odot \boldsymbol{m}_i, \boldsymbol{w}_a \odot \boldsymbol{m}_i)\\ &= \boldsymbol{x}_a \odot \boldsymbol{m}_i \oplus \boldsymbol{w}_a \odot \boldsymbol{m}_i\end{aligned} \tag{16}$$

where $\boldsymbol{\zeta}_{ai}^c$ remains the raw simple static interactive information between user a and item i.

Further, we model the dynamic and complex factors by DNNs from both social and preference perspectives

$$\boldsymbol{\eta}_{ai,t}^c = \boldsymbol{u}_{a,t}^s \oplus \boldsymbol{v}_{i,t} \oplus \boldsymbol{u}_{a,t}^p \oplus \boldsymbol{v}_{i,t} \tag{17}$$

where $\boldsymbol{\eta}_{ai,t}^c$ contains the complex and rich information for learning. Then, we refine it through DNNs as

$$\boldsymbol{h}_{ai,t}^K = \mathrm{DNN}_{K-1}^c(\ldots\mathrm{DNN}_1^c(\mathbf{Z}_1^T \cdot \boldsymbol{\eta}_{ai,t}^c) + \boldsymbol{b}_1\ldots) + \boldsymbol{b}_{K-1}. \tag{18}$$

c) Output layer: Similarly, we combine the final shallow and deep high-level embeddings to output the result

$$\hat{r}_{ai,t} = \sigma(\mathbf{Z}_K^T[\boldsymbol{h}_{ai,t}^K, \boldsymbol{\zeta}_{ai}^c] + \boldsymbol{b}_K). \tag{19}$$

d) Loss function: We adopted $l2$ loss for training, which has been widely used for recommendation models with good performance [29], then we have the following:

$$L_c = \sum_{(a,i,r_{ai,t})\in T_R} \frac{1}{2}(r_{ai,t} - \hat{r}_{ai,t})^2 \tag{20}$$

where T_R denotes all the observed user–item pairs.

C. Model Training

Finally, we combine the two kinds of loss functions for optimization

$$\begin{aligned}\min L = \lambda_s & \sum_{(a,b,y_{ab},t)\in \mathbf{Y}_t^+ \cup \mathbf{Y}_t^-} (y_{ab}\log(\hat{y}_{ab}) \\ & + (1-y_{ab})\log(1-\hat{y}_{ab})) \\ = +\lambda_c & \sum_{(a,i,r_{ai,t})\in T_R} \frac{1}{2}(r_{ai,t}-\hat{r}_{ai,t})^2 + \theta \end{aligned} \tag{21}$$

where θ is regularization parameters. Besides, to balance these two prediction tasks, we assign the tradeoff parameters λ_s and λ_c, which will play an important role in the predictions [6], we will study the influence in the next part.

V. Experiments

In this section, we conduct several experiments to answer the following three key questions, which aim to purposefully verify the effectiveness of *DGN-JBP*.

RQ1: How does *DGN-JBP* perform compared to other state-of-the-art single-task models?

RQ2: Does the feature fusion impose an influence on the performance of *DGN-JBP*?

RQ3: How do the key parameters impact the performance of *DGN-JBP*?

A. Experimental Settings

1) Dataset Description: We conduct experiments on two real-world datasets: Epinions [30] and Gowalla [48]. Epinions is a whom-trust-whom online social network platform with product sharing and social relationship information. Gowalla is a location-based social networking dataset where users share their locations through checking in. Both datasets include timestamp information, with each timestamp representing a one-month interval. During data preparation, we retained users with more than two records and normalized the rating scores to a range of [0, 1]. After that, we used data up to the timestamp $T-1$ (starting from 0) for training the model, and the data from the last timestamp was treated as new user behaviors for testing. For clarity, an overview of the detailed characteristics of two datasets after pruning is summarized in Table II.

2) Evaluation Metrics: In evaluating performance, we randomly select 100 users as negative samples for each user in the social link behavior prediction task. Then, we mix the negative samples and corresponding positive samples of each user in the test set (last timestamp). After that, the mixed data is then input into the trained model. The top-K scored candidates from the output will be recommended to the user as potential social friends, and we evaluate the performance by precision, recall, and F1 score for comparison. As for consumption behavior prediction, given that the ratings inherently include negative information, we directly calculate the RMSE on the test set for evaluation.

3) Parameters Settings: During model training, all embeddings are initialized with a Gaussian distribution, with a mean value of 0 and a standard deviation of 0.01. The range of values for the user and item embedding dimension is [5,10,15,20]. Then, when aggregating neighborhood information in GATs, we employed two layers of DNNs to train all weights, the first layer adopted a sigmoid activation function, followed by a leaky_relu activation function in the second layer. In most experiments, we maintained the number of GCN layers at 2. Following the fusion layer, a three-layer DNNs (with the activation function as tanh) is employed to extract information and output the final results. We then train the model parameters by Stochastic Gradient Descent (we adopt Adam optimizer) with mini-batch (learning rate 0.001 on both datasets). Since we have different numbers of the training set on two kinds of behavior data, we split the mini-batch by fixing the number of batches, i.e., two prediction tasks have the same number of batches with an individual batch size. For the layers of GCNs and DNNs, as well as the tradeoff parameters of the two tasks, we will discuss them in detail in the following section.

TABLE II
Statistics of the Datasets After Splitting

Dataset	Epinions	Gowalla
Users	4630	21 755
Items	26 991	71 139
Total time stamp	12	4
Training Consumption	62 872	278 154
Test Consumption	2811	52 448
Training Links	75 009	251 296
Test Links	3257	6254
Consumption Density	0.05%	0.018%
Link Density	0.35%	0.053%

B. Performance Comparison (RQ1)

1) Social Link Behavior Prediction Performance: We report the performance of NJBP model with the following social link behavior prediction methods, and the results of Epinions and Gowalla are illustrated in Figs. 4 and 5, respectively.

AA [1]: The Adamic/Adar metric is an unsupervised method for link prediction, which is based on neighborhood information.

CMF [49]: It compresses multiple temporal data into one matrix and employs matrix factorization for predictions.

hTrust [4]: This method utilizes users' preference information to improve trust prediction performance.

FIP [6]: It utilizes the same user embeddings and focuses on predicting both types of behavior.

Node2vec [17]: It is a state-of-the-art network embedding method for link prediction.

MGNN [42]: It takes advantage of GNNs to predict two kinds of behavior; nevertheless, it does not consider the temporal information.

RWJBG [8]: It is a graph embedding method utilizing random walks for two types of behavior prediction.

NJBP [7]: It is a dual-path framework that integrates shallow and deep features for modeling two behaviors.

ELJP [29]: It explores the evolution of SNSs and utilizes temporal information to predict two behaviors.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

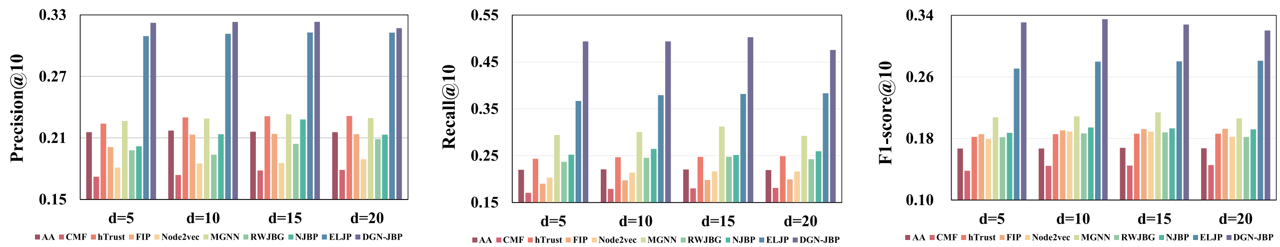

Fig. 4. Performance of social link predictions on Epinions.

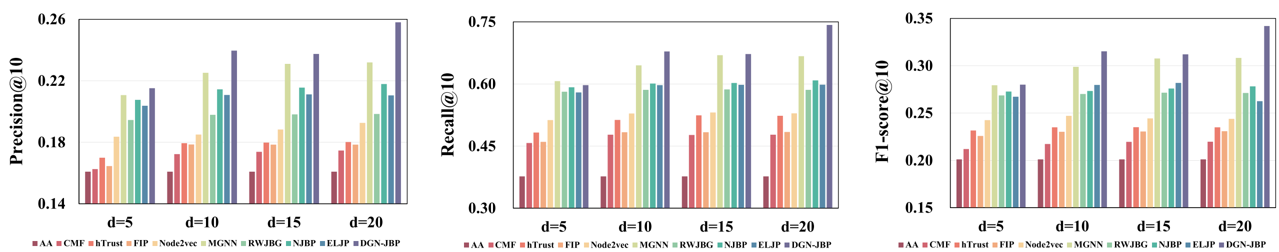

Fig. 5. Performance of social link predictions on Gowalla.

Based on the results of the two datasets, AA performs not so well compared with CMF on Epinions because it relies on a heuristic method for direct computation, and the Epinions dataset exhibits a significantly higher density compared to the Gowalla dataset. Consequently, AA is able to identify more dependable potential social neighbors by leveraging the denser social link structure. hTrust achieves slightly higher accuracy in social link prediction compared to AA and CMF, as it leverages additional user preference information. As a network embedding learning method, node2vec performs better on Gowalla than on Epinions. As mentioned earlier, the sparsity of the social network in Gowalla is significantly greater than that in Epinions. Therefore, its advantage in sparse embedding learning is more pronounced. Besides, we notice that FIP faces challenges in modeling the correlations of both behaviors when sharing the same user embeddings, particularly in capturing dynamic characteristics. Therefore, it can be observed that ELJP significantly outperforms FIP because it addresses the challenging task of capturing dynamic characteristics. Compared to ELJP, MGNN, RWJBG, and NJBP utilize information from two types of behaviors, modeling them through GNNs, network embedding learning, and deep neural networks. However, due to the inability to model dynamic characteristics, the effectiveness of these models experiences varying degrees of degradation.

2) Consumption Behavior Prediction Performance: We compare the consumption prediction results of the proposed model with the following methods.

PMF [2]: It is a traditional matrix factorization method that employs a pointwise loss function for a recommendation.

TMF [6]: It incorporates the changing dynamics of users' preferences over time to enhance CF-based models.

SocialMF [5]: It incorporates the neighborhood information with users' preferences for prediction.

ContextMF [50]: It utilizes social contextual information to get personal preferences and interpersonal influences for preference prediction.

RRN [51]: It was proposed to predict future behavioral trajectories. This is achieved by endowing both users and items with an LSTM autoregressive model that captures dynamics, and a more traditional low-rank factorization.

MGNN [42]: It takes advantage of GNNs to predict two kinds of behavior, nevertheless, it does not consider the temporal information.

RWJBG [8]: It is a graph embedding method utilizing random walks for two types of behavior prediction.

NJBP [7]: It is a dual-path framework that integrates shallow and deep features for modeling two behaviors.

ELJP [29]: It explores the evolution of SNSs and utilizes temporal information to predict two behaviors.

As shown in Figs. 6 and 7, compared to the traditional classic algorithm PMF, which solely utilizes consumption data, TMF, socialMF, and ContextMF exhibit enhanced performance due to their consideration of additional information (time and social information). The results validate the dynamic evolution characteristics of user interests and the influences from social networks. Also, we observe that, in the absence of considering social information, RRN outperforms the aforementioned methods, demonstrating the importance of capturing dynamic features for modeling user preferences. Furthermore, for the joint modeling methods, we observe that MGNN performs well even without capturing dynamic characteristics. It outperforms approaches considering time information, such as ELJP, as well as the network embedding learning method RWJBG and deep neural network method NJBP. A key reason for its success is the utilization of GNNs to capture high-order embeddings and conduct feature fusion during the joint modeling process. Finally,

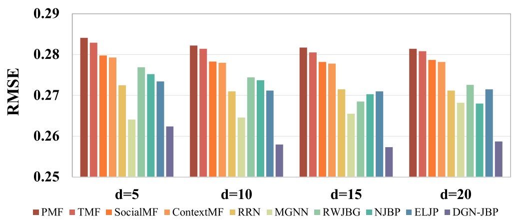

Fig. 6. Performance of consumption behavior predictions on Epinions.

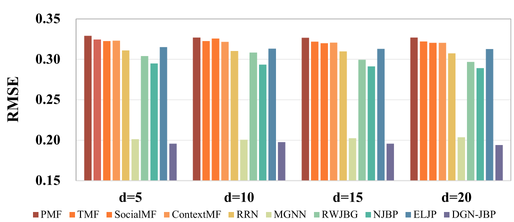

Fig. 7. Performance of consumption behavior predictions on Gowalla.

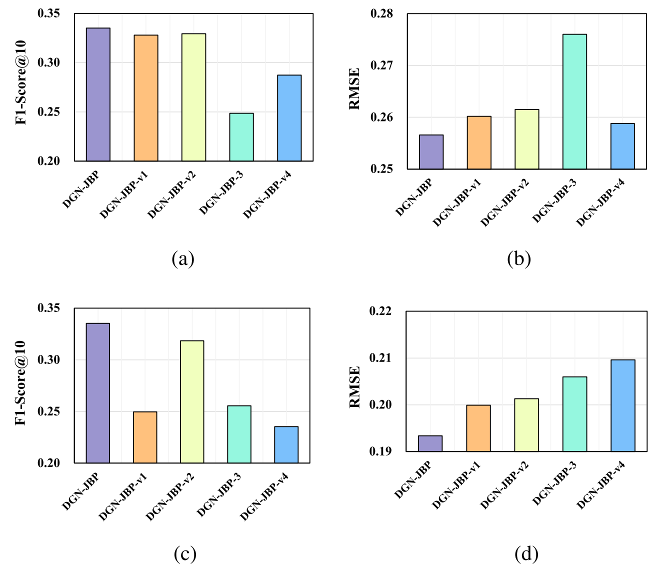

Fig. 8. Performance of variants on two datasets. (a) SLB prediction on Epinions. (b) CB prediction on Epinnions. (c) SLB prediction on Gowalla. (d) CB prediction on Gowalla.

as *DGN-JBP* addresses the shortcomings of the aforementioned methods, it achieves superior results.

C. Ablation Study (RQ2)

Since we employ embedding disentanglement by distinguishing their functions, to prove the effectiveness, we design several variants, with each one, and we remove some fusion operations. 1) In *DGN-JBP-v1*, we eliminate the preference influence in the DNFN, which means we only use user social embeddings to predict the social link behavior. Similarly, *DGN-JBP-v2* ignores the social influence in *DNPN* and only relies on deep and shallow user preference embeddings for prediction. In this way, we could prove whether the fusion operation of two tasks and information sources could promote the final results. From another perspective, as we conclude that the combination of deep and shallow embeddings can enhance the model, we discard deep embeddings in *DGN-JBP-v3* and shallow embeddings in *DGN-JBP-v4* to analyze their effectiveness for predictions. We list the results in Fig. 8, in which *DGN-JBP* achieves the best performance by taking advantage of others. "SLB" and "CB" mean social link behavior and consumption behavior, respectively, and we prove the effectiveness of the hypotheses that are included in the model.

D. Parameter Analysis

1) Diffusion Depth (RQ3): Mostly, properly increasing the convolutional layers will improve the final results; thus, the number of layers is very important as it determines the range of information aggregation in networks. Additionally, because we incorporated DNNs in the model to refine embeddings, the number of DNN layers similarly affects the results. Therefore, in this experiment, we set the number of GCN and DNN layers to a maximum of 3. There are several reasons for this consideration: many studies have shown that the majority of user relationships do not extend too far in a social network [26], [41], and preferences do not diffuse too far away either [41]. 2) Frequent convolution operations will lead to feature smoothing; to mitigate this impact, we have concatenated the embeddings from different layers, which is a technique widely adopted by other works [52]. On the contrary, a small number of convolutional layers makes it challenging to collect sufficient information. 3) Similarly, when determining the number of layers in DNNs, having too many layers may introduce additional parameters. For DNNs used in joint behavior prediction or single-task prediction, it is reasonable to set the number of layers to be within three layers [7], [9]. For this purpose, we study these influences on both datasets, and the results are shown in Table III. From the experimental results, we can draw two conclusions: 1) increasing the number of GCN layers appropriately leads to a performance improvement to some extent; and 2) however, the increase in the number of DNN layers shows unstable results, and its outcome is influenced by the number of GCN layers.

2) The Effectiveness of Attention Network (RQ3): In this section, we discuss the effects of the attention mechanisms in *DGN-JBP*. The neighborhood attention flexibly collects K-hop information by assigning adjustable weights to each link. The impact of the attention mechanism is illustrated in Fig. 9, where "AVG" indicates that the attention is not set up. The results of behavior prediction on both datasets consistently demonstrate a certain performance improvement with the introduction of the attention mechanism.

3) Social Influence (RQ3): Regarding the weight parameters that combine two tasks, the weight parameter λ_s for the social link prediction task, in a certain sense, represents the degree of social influence. We do not assign trainable weights for the two tasks to automatically adjust the weights. The purpose is to observe the trend changes and investigate the impact on the results. In the experiments, we analyzed the impacts of social

TABLE III
IMPACT OF VARYING THE NUMBER OF LAYERS ON TWO DATASETS

	Epinions						Gowalla					
	DNN = 1		DNN = 2		DNN = 3		DNN = 1		DNN = 2		DNN = 3	
	F1@10	RMSE	F1@10	RMSE	F1@10	RMSE	F1@10	RMSE	F1@10	RMSE	F1@10	RMSE
GCN = 1	0.3272	0.2572	0.3252	0.2581	0.3301	0.2582	0.3006	0.1927	0.3027	0.1930	0.3278	0.1934
GCN = 2	0.3296	0.2560	0.3254	0.2597	0.3323	0.2580	0.3353	0.1934	0.3173	0.1939	0.3162	0.1957
GCN = 3	0.3115	0.2619	0.3157	0.2624	0.3265	0.2573	0.2939	0.1979	0.3061	0.1973	0.3102	0.1961

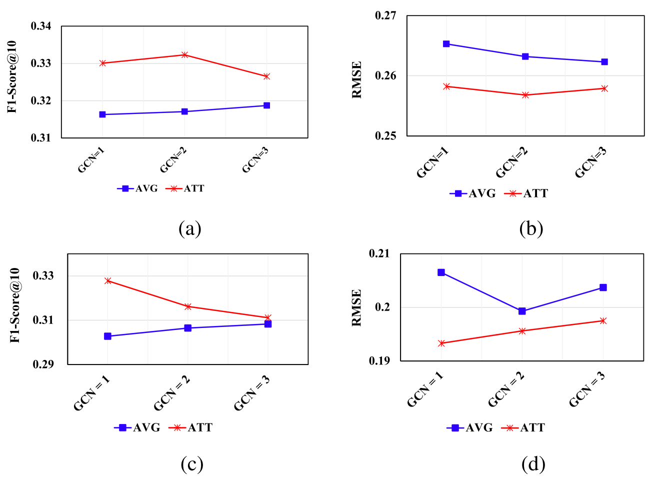

Fig. 9. Influence of attention on two datasets. (a) SLB prediction on Epinions. (b) CB prediction on Epinnions. (c) SLB prediction on Gowalla. (d) CB prediction on Gowalla.

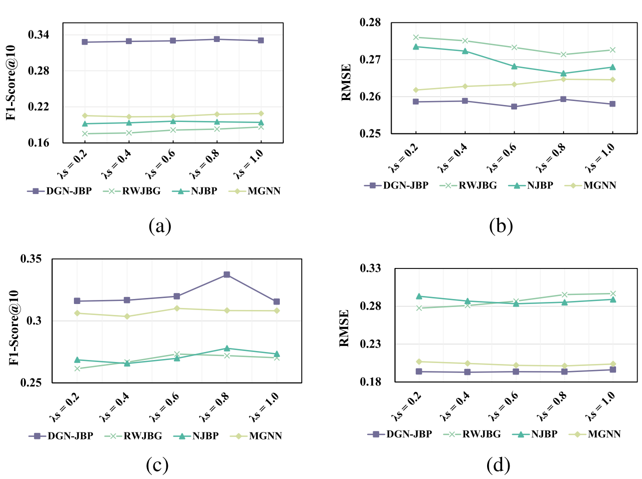

Fig. 10. Impact of social influence on two datasets. (a) SLB prediction on Epinions. (b) CB prediction on Epinnions. (c) SLB prediction on Gowalla. (d) CB prediction on Gowalla.

influence and conducted experiments by varying λ_s from 0.2 to 1.0 (with intervals of 0.2) for observation.

The results in Fig. 10 have already demonstrated that the social factor can be useful in stimulating consumption behavior. However, assigning excessive weight to it can lead to a decline in performance. The conclusion is also consistent with some similar studies [6], [7], i.e., social influence plays a positive role in promoting user consumption behavior on SNSs. However, excessive influence can also detrimentally affect platform performance. Also, due to the interaction of the embedding fusion in the model, the performance of social link and consumption predictions will be improved compared to any single prediction.

VI. CONCLUSION

In this article, we presented a new unified graph neural-based framework named DGN-JBP model, which could capture the dynamic characteristics of these two behaviors, aggregate the higher order information among graphs, and model the complex interactions between behaviors for users' two kinds of behaviors prediction. Specifically, we designed an attention neural network to flexibly aggregate disentangled user embeddings with multiperspectives from created user behavior tensors. These embeddings were then combined with individual GRUs to extract dynamic characteristics. After that, as there are mutual correlations and facilitation between two behaviors, we proposed a dual framework with two components corresponding to each task, i.e., *DNFN* and *DNPN*, allowing them to mutually enhance each other. Finally, we conducted a comprehensive set of experiments on two real-world datasets containing time stamp information. Besides comparing it with various baselines for each type of behavior to evaluate the overall performance, we also conducted a series of auxiliary experiments to validate it from multiple perspectives. In the future, as the prediction results of user behavior in current works lack modeling analysis of motive patterns and interpretation of behavior outcomes, we will continue to explore and expand the current model in this aspect.

REFERENCES

[1] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," in *Proc. 12th Int. Conf. Inf. Knowl. Manage.*, Nov. 2003, pp. 556–559.
[2] A. Mnih and R. R. Salakhutdinov, "Probabilistic matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 20, 2007.
[3] S. Aral, L. Muchnik, and A. Sundararajan, "Distinguishing influence-based contagion from homophily-driven diffusion in dynamic networks," *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 51, pp. 21544–21549, Dec. 2009.
[4] J. Tang, H. Gao, X. Hu, and H. Liu, "Exploiting homophily effect for trust prediction," in *Proc. 6th ACM Int. Conf. Web Search Data Mining*, Feb. 2013, pp. 53–62.
[5] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proc. 4th ACM Conf. Recommender Syst.*, Sep. 2010, pp. 135–142.
[6] S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha, "Like like alike: Joint friendship and interest propagation in social networks," in *Proc. 20th Int. Conf. World Wide Web*, Mar. 2011, pp. 537–546.

[7] J. Li, L. Wu, R. Hong, K. Zhang, Y. Ge, and Y. Li, "A joint neural model for user behavior prediction on social networking platforms," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 6, pp. 1–25, Sep. 2020.

[8] J. Li, L. Wu, R. Hong, and J. Hou, "Random walk based distributed representation learning and prediction on social networking services," *Inf. Sci.*, vol. 549, pp. 328–346, Mar. 2021.

[9] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proc. 26th Int. Conf. World Wide Web*, Apr. 2017, pp. 173–182.

[10] A. Kumar, S. S. Singh, K. Singh, and B. Biswas, "Link prediction techniques, applications, and performance: A survey," *Physica A*, vol. 553, Sep. 2020, Art. no. 124289.

[11] D. Lin et al., "An information-theoretic definition of similarity," in *Proc. 15th Int. Conf. Mach. Learn.*, vol. 98, Jul. 1998, pp. 296–304.

[12] E. A. Leicht, P. Holme, and M. E. Newman, "Vertex similarity in networks," *Phys. Rev. E*, vol. 73, no. 2, Feb. 2006, Art. no. 026120.

[13] G. Jeh and J. Widom, "SimRank: A measure of structural-context similarity," in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2002, pp. 538–543.

[14] A. K. Menon and C. Elkan, "Link prediction via matrix factorization," in *Proc. Eur. Conf. Mach. Learn. Knowl. Discovery Databases (ECML PKDD)*, 2011, pp. 437–452.

[15] L. Liao, X. He, H. Zhang, and T.-S. Chua, "Attributed social network embedding," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 12, pp. 2257–2270, Mar. 2018.

[16] X. Cao, H. Chen, X. Wang, W. Zhang, and Y. Yu, "Neural link prediction over aligned networks," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 249–256.

[17] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 855–864.

[18] J. Yu, H. Yin, X. Xia, T. Chen, J. Li, and Z. Huang, "Self-supervised learning for recommender systems: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 1, pp. 335–355, Jan. 2024.

[19] H. Ko, S. Lee, Y. Park, and A. Choi, "A survey of recommendation systems: Recommendation models, techniques, and application fields," *Electronics*, vol. 11, no. 1, Jan. 2022, Art. no. 141.

[20] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Comput. Surv.*, vol. 52, no. 1, Feb. 2019, Art. no. 5.

[21] L. Wu, Y. Yang, K. Zhang, R. Hong, Y. Fu, and M. Wang, "Joint item recommendation and attribute inference: An adaptive graph convolutional network approach," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, Jul. 2020, pp. 679–688.

[22] L. Wu, X. He, X. Wang, K. Zhang, and M. Wang, "A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 5, pp. 4425–4445, Jan. 2022.

[23] L. Wu, L. Chen, R. Hong, Y. Fu, X. Xie, and M. Wang, "A hierarchical attention model for social contextual image recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 10, pp. 1854–1867, Oct. 2020.

[24] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annu. Rev. Sociol.*, vol. 27, no. 1, pp. 415–444, Aug. 2001.

[25] Y. Leng, L. Yu, and X. Niu, "Dynamically aggregating individuals' social influence and interest evolution for group recommendations," *Inf. Sci.*, vol. 614, pp. 223–239, Oct. 2022.

[26] L. Wu, P. Sun, Y. Fu, R. Hong, X. Wang, and M. Wang, "A neural influence diffusion model for social recommendation," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, Jul. 2019, pp. 235–244.

[27] G. Guo, J. Zhang, and N. Yorke-Smith, "TrustSVD: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 123–129.

[28] X. Sha, Z. Sun, and J. Zhang, "Disentangling multi-facet social relations for recommendation," *IEEE Trans. Comput. Soc. Syst.*, vol. 9, no. 3, pp. 867–878, Jun. 2022.

[29] L. Wu et al., "Modeling the evolution of users' preferences and social links in social networking services," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 6, pp. 1240–1253, Jun. 2017.

[30] P. Wu, Y. Tu, X. Yuan, A. Jatowt, and Z. Yang, "Neural framework for joint evolution modeling of user feedback and social links in dynamic social networks," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 1632–1638.

[31] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. 5th Int. Conf. Learn. Representations*, 2017.

[32] Q. He, J. Chen, H. Xu, and K. He, "Structural robust label propagation on homogeneous graphs," in *Proc. 22nd Int. Conf. Data Mining*, 2022, pp. 181–190.

[33] Y. Sun, D. Zhu, H. Du, and Z. Tian, "MHNF: Multi-hop heterogeneous neighborhood information fusion graph representation learning," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 7, pp. 7192–7205, Jun. 2022.

[34] X. Gong, C. P. Chen, and T. Zhang, "Cross-cultural emotion recognition with EEG and eye movement signals based on multiple stacked broad learning system," *IEEE Trans. Comput. Soc. Syst.*, vol. 11, no. 2, pp. 2014–2025, Aug. 2023.

[35] X. Gong, C. P. Chen, B. Hu, and T. Zhang, "CiABL: Completeness-induced adaptative broad learning for cross-subject emotion recognition with EEG and eye movement signals," *IEEE Trans. Affect. Comput.*, early access, Apr. 23, 2024.

[36] Y. Yang et al., "Hyperbolic graph learning for social recommendation," *IEEE Trans. Knowl. Data Eng.*, Dec. 15, 2023.

[37] P. Shao et al., "Average user-side counterfactual fairness for collaborative filtering," *ACM Trans. Inf. Syst.*, vol. 42, no. 5, pp. 1–26, May 2024.

[38] M. Liu, H. Gao, and S. Ji, "Towards deeper graph neural networks," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 338–348.

[39] L. Chen, L. Wu, R. Hong, K. Zhang, and M. Wang, "Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach," in *Proc. 34th AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 1, pp. 27–34.

[40] W. Fan et al., "Graph neural networks for social recommendation," in *Proc. 28th Int. Conf. World Wide Web*, May 2019, pp. 417–426.

[41] L. Wu, J. Li, P. Sun, R. Hong, Y. Ge, and M. Wang, "DiffNet++: A neural influence and interest diffusion network for social recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 10, pp. 4753–4766, Oct. 2022.

[42] Y. Xiao, L. Yao, Q. Pei, X. Wang, J. Yang, and Q. Z. Sheng, "MGNN: Mutualistic graph neural network for joint friend and item recommendation," *IEEE Intell. Syst.*, vol. 35, no. 5, pp. 7–17, Apr. 2020.

[43] F. Manessi, A. Rozza, and M. Manzo, "Dynamic graph convolutional networks," *Pattern Recognit.*, vol. 97, Jan. 2020, Art. no. 107000.

[44] S. Wu, Y. Tang, Y. Zhu, L. Wang, X. Xie, and T. Tan, "Session-based recommendation with graph neural networks," in *Proc. 33rd AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 1, pp. 346–353.

[45] J. Chen, X. Wang, and X. Xu, "GC-LSTM: Graph convolution embedded LSTM for dynamic network link prediction," *Appl. Intell.*, vol. 52, pp. 7513–7528, Sep. 2021.

[46] T. Bai, J.-R. Wen, J. Zhang, and W. X. Zhao, "A neural collaborative filtering model with interaction-based neighborhood," in *Proc. 26th ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 1979–1982.

[47] H.-J. Xue, X. Dai, J. Zhang, S. Huang, and J. Chen, "Deep matrix factorization models for recommender systems," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 3203–3209.

[48] S. Scellato, A. Noulas, and C. Mascolo, "Exploiting place features in link prediction on location-based social networks," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2011, pp. 1046–1054.

[49] D. M. Dunlavy, T. G. Kolda, and E. Acar, "Temporal link prediction using matrix and tensor factorizations," *ACM Trans. Knowl. Discovery Data*, vol. 5, no. 2, pp. 1–27, Feb. 2011.

[50] M. Jiang, P. Cui, F. Wang, W. Zhu, and S. Yang, "Scalable recommendation with social contextual information," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 11, pp. 2789–2802, Nov. 2014.

[51] C.-Y. Wu, A. Ahmed, A. Beutel, A. J. Smola, and H. Jing, "Recurrent recommender networks," in *Proc. 10th ACM Int. Conf. Web Search Data Mining*, Feb. 2017, pp. 495–503.

[52] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," in *Proc. 28th Int. Conf. World Wide Web*, Jul. 2019, pp. 417–426.

Junwei Li received the Ph.D. degree in information and communication engineering from the School of Computer Science and Information Engineering, Hefei University of Technology, Hefei, China, in 2021.

He is currently a Faculty Member with Chongqing University of Posts and Telecommunications, Chongqing, China. His research interests include user behavior analysis and recommender systems. He has published several papers in refereed journals and conferences, such as ACM Transactions on Intelligent Systems and Technology, and IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING.

Le Wu received the Ph.D. degree in technology for computer applications from the University of Science and Technology of China, Hefei, China.

She is currently a Professor with Hefei University of Technology, Hefei, China. Her research interests include data mining, recommender systems, and social network analysis. She has published more than 40 papers in referred journals and conferences.

Dr. Wu was the recipient of the Best of SDM 2015 Award, and the Distinguished Dissertation Award from China Association for Artificial Intelligence (CAAI) 2017.

Yulu Du received the Ph.D. degree in computer science from Beijing University of Posts and Telecommunications, Beijing, China, in 2019.

He is currently a Lecturer with Chongqing University of Posts and Telecommunications, Chongqing, China. His research interests include recommender systems and intelligent information processing. He has published several papers in refereed journals and conferences, such as IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING (TKDE) and *Information Sciences*.

Richang Hong (Senior Member, IEEE) received the Ph.D. degree in signal and information processing from the University of Science and Technology of China (USTC), Hefei, China, in 2008.

He is currently a Professor with Hefei University of Technology (HFUT), Hefei, China. He has co-authored over 70 publications in the areas of his research interests that include multimedia question answering, video content analysis, and pattern recognition.

Dr. Hong is a Member of the Association for Computing Machinery. He was a recipient of the Best Paper award in the ACM Multimedia 2010.

Weisheng Li (Member, IEEE) received the B.S. and M.S. degrees from the School of Electronics and Mechanical Engineering and the Ph.D. degree from the School of Computer Science and Technology, Xidian University, Xi'an, China, in 1997, 2000, and 2004, respectively.

He is currently a Professor with Chongqing University of Posts and Telecommunications, Chongqing, China. His research interests include intelligent information processing and pattern recognition.