DREAM: A Dynamic Relational-Aware Model for Social Recommendation

Liqiang Song, Ye Bi, Mengqiu Yao, Zhenyu Wu, Jianming Wang, Jing Xiao

magicyebi@163.com,{songliqiang537,yaomengqiu621,wuzhenyu447,wangjianming888,xiaojing661}@pingan.com.cn

Ping An Technology Shenzhen Co., Ltd

ABSTRACT

Social connections play a vital role in improving the performance of recommendation systems (RS). However, incorporating social information into RS is challenging. Most existing models usually consider social influences in a given session, ignoring that both usersâĂŹ preferences and their friendsâĂŹ influences are evolving. Moreover, in real world, social relations are sparse. Modeling dynamic influences and alleviating data sparsity is of great importance.

In this paper, we propose a unified framework named Dynamic RElation-Aware Model (DREAM) for social recommendation, which tries to model both usersâĂŹ dynamic interests and their friendsâĂŹ temporal influences. Specifically, we design temporal information encoding modules, because of which user representations are updated in each session. The updated user representations are transferred to relational-GAT modules, subsequently influence the operations on social networks. In each session, to solve social relation sparsity, we utilize glove-based method to complete social network with virtual friends. Then we employ relational-GAT module over completed social networks to update usersâĂŹ representations. In the extensive experiments on the public datasets, DREAM significantly outperforms the state-of-the-art solutions.

CCS CONCEPTS

• **Applied computing** → **Online shopping**; • **Networks** → *Online Social Networks*.

KEYWORDS

Session-based Social Recommendation, Virtual Friends, Temporal Information Encoding

1 INTRODUCTION

Recommendation systems (RS), attempting to help users overcome information overload, have become popular. RS make use of users' historical activities and/or social relationships to generate features of items and identify users' latent preferences. Among RS strategies, collaborate filtering (CF)-based methods have received significant success [7]. However, CF-based methods usually suffer from sparsity of user-item interactions and cold start problem.

To address these limitations, modern RS also generate other useful data. By combining rating matrices with additional data, better recommendations are respected. Nowadays, the increasing popularity of social media greatly enriches people's social activities, which harnesses social relations to boost the performance of RS [1]. Based on the intuition that people in same social group are likely to have similar preferences, and that users will gather information from their social friends, lots of work has incorporated social information into RS, e.g. SBPR [11], GraphRec [1], etc..However, incorporating social information into RS is challenging.

In general, methods utilizing heterogeneous data tend to perform better than those using a single data source [3]. DGRec [8] models dynamic user behaviors with RNN, and social influence with GAT, and is proven to outperform the state-of-the-art solutions. However, the work only considers social influences in a given session, ignoring the effects of temporal dependency among different sessions. Besides, relying only on social friends is far from satisfactory, since in real world, the social relations are very sparse.

Based on the observations, we propose a novel framework called A Dynamic RElational-Aware Model (DREAM), which tries to model both usersâĂŹ dynamic individual interests and their friendsâĂŹ temporal influences. The model contains two main modules, temporal information encoding (TIE) modules encode the outputs form historical sessions by recursively combining the representation of all previous outputs with the current output. In each session, we first complete social network, and then employ relational-aware graph attention (relational-GAT) network modules to integrate influences from users' real and virtual friends. To be specific, we first design a method to increase the number of friends, which refer to the neighbors whose behaviors are similar to the users, and we call this kind of neighbors as virtual friends. Within each session, we then utilize relational-GAT [10] to capture the influences of friends. As we know, user preference for products drifts over time. To model users' dynamic preferences and the dynamic influences from their friends, we design the temporal information encoding modules. In each TIE module, we combine the relational-GAT encoded features as well as the features from last TIE module. In summary, our contributions in this paper are as follows:

- We propose a novel RS approach, which aims to model usersâĂŹ dynamic interests and dynamic influences from

their friends. The model encodes the outputs form historical sessions by recursively combining the features encoded by relational-GAT modules and that from last TIE module.

- We design a <u>GloVe-based method to increase the number of friends,</u> and use relational-GAT to aggregate user representations from both completed social network in each session.
- We conduct experiments on real-world recommendation scenarios, and the results prove the efficacy of DREAM over several state-of-the-art baselines.

2 PROBLEM FORMULATION

Let $\mathcal{U} = \{u_1, u_2, \ldots u_n\}$ and $\mathcal{I} = \{i_1, i_2, \ldots i_m\}$ denote the sets of users and items, where n and m are the number of users and items, respectively. The user-item interaction matrix $Y \in \mathbb{R}^{n \times m}$ is defined as: $y_{u,i} = 1$, if there is an interaction between u and i, and 0 otherwise. Each u is associated with a set of sessions, $\mathcal{I}_T^u = \{\mathcal{S}_1^u, \mathcal{S}_2^u, \ldots, \mathcal{S}_T^u\}$, where $\mathcal{S}_t^u$ is the t-th session of user u, each session $\mathcal{S}_t^u$ consists of a user behaviors sequence $\{i_{t,1}^u, i_{t,2}^u, \ldots, i_{t,N_{u,t}}^u\}$, where $i_{t,p}^u$ is the p-th item interacted by user u in t-th session. Social network can be described by $S^R \in \mathbb{R}^{n \times n}$, where $s_{p,q}^R = 1$ if there is a relation between u_p and u_q, and 0 otherwise. We call embedding vectors $\boldsymbol{u}$ and $\boldsymbol{i}$ latent feature vectors. Given Y and S^R, we aim to predict whether u has potential interests in target item v.

3 DREAM FRAMEWORK

The framework is illustrated in Figure 1. The model consists of social network completion, relational-aware graph attention network (relational-GAT) modules, temporal information encoding (TIE) modules and recommendation. Given a user u, we first select her T historical sessions. In t-th session, we first complete social network, and get $\mathcal{G}_t^C$. The inputs of the model are graph sequence $\{\mathcal{G}_1^C, \mathcal{G}_2^C, \ldots, \mathcal{G}_T^C\}$. In each session, relational is employed to integrate influences from $\mathcal{G}_t^C$. User representations are updated in TIE module and transferred to next relational module.

3.1 Social Network Completion

Virtual Friends Definition and Selection. We define *virtual friend* as users having similar consumption habits, and the connection is stronger if they are more similar. Under the definition, we design a GloVe-based method. We first utilize GloVe mechanism [6] to learn user representations, and then calculated the similarity among all users, finally we choose top-k users whose similarity is higher. The input is user-user co-occurrence counts matrix X, whose entries $X_{p,q}$ denote the number of times user p and user q consume the same items. And outputs are user embeddings $\boldsymbol{g}_u$. We define the connection strength between user u_p and u_q as:

$$s_{p,q}^V = \text{softmax}(\langle \boldsymbol{g}_{u_p}, \boldsymbol{g}_{u_q} \rangle) = \frac{\exp(\langle \boldsymbol{g}_{u_p}, \boldsymbol{g}_{u_q} \rangle)}{\sum_{u_l, u_s \in \mathcal{U}} \exp(\langle \boldsymbol{g}_{u_l}, \boldsymbol{g}_{u_s} \rangle)},$$

and u_p's virtual friends set is defined as $\mathcal{N}^V(u_p)$.

Node Representation. Friends' influences always lag, since friends may consumed products first and then influence the user. So in t-th relational-GAT module, friends node representations are calculated in $(t$-1)-th session. Here, we use users' short-term interests,
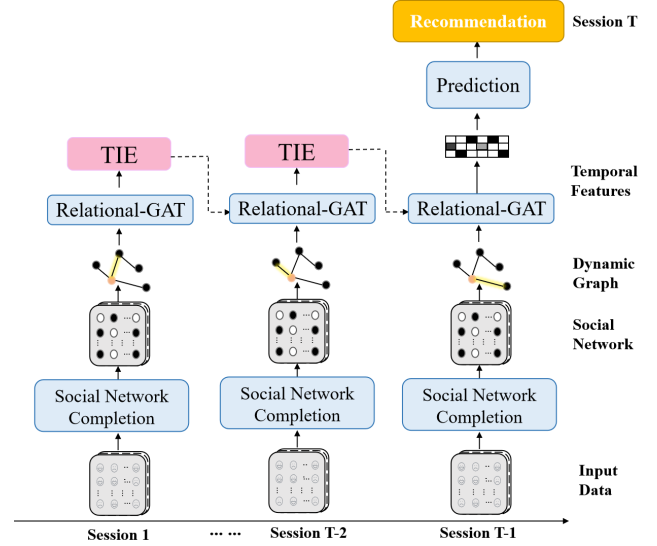

Figure 1: DREAM Framework. In each session, relational is employed to integrate influences from completed social network. User representations are updated in TIE module and transferred to next relational-GAT module.

which are gotten by employing GRU [2] over user u_j's interaction sequence in $(t$-1)-th session $\mathcal{S}_{t-1}^j = \{i_{t-1,1}^j, i_{t-1,2}^j, \ldots, i_{t-1,N_{k,t}}^j\}$, i.e. $\boldsymbol{s}_j = \text{GRU}(\mathcal{S}_{t-1}^j)$.

3.2 Relational-GAT Module

Note there are two different relations in our completed social network, to capture this difference, we employ relational-GAT over the completed social network in each session. In t-th session, for each user, we build a graph where nodes correspond to target user and her friends. For target user u with $|\mathcal{N}(u)|$ friends, the graph has $|\mathcal{N}(u)| + 1$ nodes. we rerepresent nodes as $\boldsymbol{h}_u^{(0)} = \widetilde{\boldsymbol{u}}_{t-1}$ and $\{\boldsymbol{h}_j^{(0)} = \boldsymbol{s}_j\}$, where user node representation $\widetilde{\boldsymbol{u}}_{t-1}$ is gotten from last TIE module (see section 3.3). Then we employ relational-GAT over $\mathcal{N}(u)$. Specifically, we first project friends to the same space, and then calculate the attention score:

$$\alpha_{uk} = \frac{\exp(f_r(\boldsymbol{h}_u^{(0)}, P_r \boldsymbol{h}_k^{(0)}))}{\sum_{u_j \in \mathcal{N}(u) \cup \{u\}} \exp(f_r(\boldsymbol{h}_u^{(0)}, P_r \boldsymbol{h}_j^{(0)}))}, \quad \forall w \in \mathcal{N}(u),$$

where $f_r(\cdot, \cdot)$ is the deep neutral network performing relational attention. Then, we aggregate information from $\mathcal{N}(u)$:

$$\boldsymbol{h}_u = \sigma \left(\sum_{u_j \in \mathcal{N}(u) \cup \{u\}} \alpha_{uj} \boldsymbol{h}_j^{(0)} \right),$$

where σ denotes the activation function. We denote the final representation of user as $\boldsymbol{u}_t = \boldsymbol{h}_u$.

3.3 Temporal Information Encoding Module

Temporal information is an important factor and there are traditional RS that consider temporal information [8]. However, exploiting temporal personal preference and dynamic influences from

users' friends together is still challenging. For example, traditional RNN usually performs worse as the length of the sequences increases. Besides in the real world, the influences produced in the earlier session may decay as time going by. Inspired by Gated Recurrent Unit (GRU) [2] we propose the temporal information encoding modules. As shown in Figure 1, we first select usersâĂŹ historical behavior sessions, and utilize temporal information encoding modules among the outputs from each session to grasp the dynamic interests and influences. The output of relational-GAT module in session t is denoted as u_t. To combine relational-GAT encoded features at each time, as well as the target user's dynamic personal interests, we design a GRU-like module, called TIE module. The hidden layer of TIE is a linear interpolation between the last TIE's hidden layer $\widetilde{u}_{t-1}$ and the candidate hidden layer $\widetilde{h}_t$. The encoding procedures are defined as below:

$$u_q = W_q^t \widetilde{u}_{t-1} + b_q^t$$
$$u_e = W_e^t u_t + b_e^t$$
$$\widetilde{h}_t = \tanh(W_h^t u_t + u_e \circ U_h^t \widetilde{u}_{t-1} + b_h^t)$$
$$\widetilde{u}_t = (1 - u_q) \circ \widetilde{u}_{t-1} + u_q \circ \widetilde{h}_t,$$

where $W_q^t, W_e^t, W_h^t, U_h^t \in \mathbb{R}^{d \times d}, b_q^t, b_e^t, b_h^t \in \mathbb{R}^d$. We let $\widetilde{u}_0 = u$, the initial user latent feature u. So the vector $\widetilde{u}_t$ can be seen as user's long-term preference combining her evolving interests and the dynamic influences from her friends.

3.4 Prediction

The final output of relational-GAT module u_T is used as the target user representation, then the user representation u_T and the target item v are combined to predict the clicking probability $\hat{y}_{uv} = \sigma(f(u_T, v))$, where $\sigma(\cdot, \cdot)$ is sigmoid function, and f is a ranking function which can be either a dot-product function or a deep neural network. The loss function L is the sigmoid cross entropy loss:

$$\sum_{(u,v) \in R} -(y_{uv} \log \sigma(f(u_T, v)) + (1 - y_{uv}) \log(1 - \sigma(f(u_T, v))))$$

4 EXPERIMENT

To comprehensively study our proposed model DREAM, we conduct experiments on real-world datasets to answer the following questions: Q1: Does DREAM outperform the state-of-the-art baselines? Q2: How is DREAM affected by each component?

Table 1: Descriptive statistics of two datasets

	Epinions	Movie
# Users	15,489	57,496
# Items	255,253	56,858
# Events	500,770	3,007,442
# Social links	355,217	1,758,302
Avg.session/user	2.3778	2.8786
Avg. real friends/user	22.99	15.60

4.1 Experimental Setup

Datasets. *Epinions*[1] and *Douban-Movie*[2] (Short for Movie) are utilized to evaluate the performance of DREAM. As for Epinions, "trust" endorsements are converted to directed edges in social network and users' behaviors are segmented into *month-long* sessions. As for Douban-Movie, we crawled the interaction data using the identities of users in movie community along with associated timestamps and usersâĂŹ social netowrks. We construct our datasets by using each review as an evidence that a user consumed an item and segment usersâĂŹ behaviors into *week-long* sessions for their high activeness.We convert usersâĂŹ explicit ratings on items into 1 as the implicit feedback for all the datasets. We randomly split the user-item interactions of each dataset into training set (80%) to learn the parameters, validation set (10%) to tune hyper-parameters, and testing set (10%) for the final performance comparison. The statistics of two datasets are presented in Table 1.

Baselines. In this subsection, we compare DREAM with four groups of recommendation baselines. (1) only considers user feedbacks: BPR[7]. (2) considers social network information: GraphRec [1] and SBPR [11]. (3) considers the sequence of user actions: GRU [2] and SASRec [4]. (4) combines social network and user actionsâĂŹ temporal information into recommendation task: DGRec [?].

We also construct several variants of DREAM for ablation study on two datasets. There are two special changes in our DREAM. One is the addition of virtual friendsâĂŹ information in social network, the other change is the utilization of more than one session information (the best parameter is 2). As for inner-session information, we design two variants of DREAM, DREAM-R and DREAM-V to evaluate the effect of virtual friends' information. DREAM-R removes virtual friends' information, while DREAM-V uses virtual friends' information only. As for inter-session information, DREAM-GAT utilizes GAT[9] to combine two friend-level information rather than realtional-GAT. And,DREAM-TGRU leverages GRU to fuse two session information to evaluate the necessity of TIE module. DREAM-1 uses only one sessionâĂŹs information like DGRec, DREAM-3 utilizes 3 sessions' information.

Metrics. R@K (Short for Recall@K,K=10), NDCG and MRR are utilized to measure the performance. For all the metrics, the larger the values, the better the performance. In order to reduce the computational since there are too many unrated items, we randomly sample 1000 unrated items as negative samples and combine them with positive items in the ranking process for each user. We repeat this procedure 10 times and report the average ranking results.

Parameter Setting. For all the models that are based on the latent factor models, we initialize the latent vectors with small random values. We use Adam [5] as the optimizing method for all models that relied on the gradient descent based methods with a learning rate of 0.0001 and batch size of 32. In DREAM model, GloVe algorithm used to find high-quality virtual friends gets the best performance compared to rating matrix. The real and virtual friends sampling numbers are uniformly set to 10 after lots of experiments. For relational-GAT module in DREAM, we set the corresponding

parameters according to their original paper [10]. The regularization parameter as 0.00001 and ReLU is utilized to implement the non-linear transformation function. In order to make the models converge faster, we apply the batch normalization. The best models are selected by early stopping when the validation accuracy does not increase for 5 consecutive epochs. There are several other parameters in the baselines, we tune all these parameters to ensure the best performance of the baselines for fair comparison.

4.2 Model Analysis

Overall Performance: Q1. As shown in Table 2, the following observations can be made: SBPR and GraphRec utilize both user-item interactions and social relations: while BPR only uses user-item interactions. SBPR and GraphRec outperform BPR, which is consistent with previous work. This indicates that social information reflects usersâĂŹ interests effectively.

Table 2: Overall performance. âĂŸImprv.âĂŹ denotes percentage improvement of DREAM, with respect to the best baseline.

Model	Epinions			Movie		
	R@10	NDCG	MRR	R@10	NDCG	MRR
BPR	0.00585	0.08396	0.00228	0.01574	0.11265	0.00651
SBPR	0.00658	0.08948	0.00281	0.01642	0.11333	0.00685
GraphRec	0.00880	0.09635	0.00409	0.01787	0.11352	0.00698
GRU	0.00410	0.09229	0.00360	0.01141	0.11380	0.00700
SASRec	0.00410	0.09239	0.00287	0.01723	0.11459	0.00747
DGRec	**0.01176**	**0.09632**	**0.00468**	**0.01901**	**0.11486**	**0.00750**
DREAM	0.01639	0.09787	0.00628	0.02285	0.11669	0.00870
Imprv.	39.37%	1.58%	34.19%	20.20%	1.59%	16.00%

In most cases, GRU and SASRec obtain better performance than BPR, which are modeled with temporal information. These improvements reflect the power of temporal information on RS. However, the performance of methods of social relations sometimes exceed temporal information based methods, which suggests it is hard to determine which is better. DGRec combining social relations and temporal information achieves much better performance than classic, social and temporal baselines. DGRec utilizes dynamic interests and static interests social information to represent target userâĂŹs interests, meanwhile, it employs graph-based algorithm with attention mechanism to deal with the influence of social friends, which shows the effectiveness of these two changes.

Our model DREAM consistently outperforms all the baselines. The substantial improvement of DREAM over the baselines could be attributed to three reasons: 1) We expand target userâĂŹs social network using virtual friends, which comprehensively expresses target userâĂŹs dynamic and static interests. 2) We combine userâĂŹs historical representation and current representation as the input of TIE. Utilizing updated user representation considers is the important step to learn the evolution of target userâĂŹs interests. 3) Different from DGRec only using one session information, we design TIE module to employ multiple temporal sessionsâĂŹ information which reflects the evolution of target userâĂŹs dynamic interests over time. This strongly indicates the advantage of using temporal information across sessions. In terms of time complexity, we find that DREAM using 2-session information (25 seconds) increases 1.9 seconds than DGREC (23.1 seconds) by calculating running time spent on single GPU per epoch.

Ablation Study: RQ2. Table 3 are the ablation study's results of DREAM in terms of R@10 and MRR. For inner-session aspect, we compare DREAM with DREAM-R and DREAM-V. DREAM-V captures users' dynamic interest information hidden in graph structure, while DREAM-R makes up corresponding social information. So, DREAM using both real and virtual friends' information gets better performance than DREAM-R and DREAM-V. In inter-session aspect, we also evaluate the necessity of relational-GAT dealing with differernt kinds of social information and TIE module. The performance of DREAM-GAT is slightly poorer than ours, indicating relational-GAT can sharply catch the difference between the two social information. The results show that TIE module improves the performance compared with DREAM-TGRU. This fully verifies the temporal effects in capturing users' interests. More importantly, the more session information doesnâĂŹt mean better performance. DREAM utilizing 2 sessionâĂŹs information gets better performance than DREAM-3. Moreover, the more session information used, the longger running time it takes to run.

Table 3: Ablation study comparing the performance of the complete model DREAM with several variations.

Model Components		Epinions		Movie	
		R@10	MRR	R@10	MRR
Inner-Session	DREAM-R	0.00820	0.00325	0.01868	0.00759
	DREAM-V	0.01230	0.00347	0.01873	0.00765
Inter-Session	DREAM-GAT	0.01510	0.00527	0.02109	0.00816
	DREAM-TGRU	0.01530	0.00551	0.02186	0.00837
	DREAM-S1	0.01297	0.00389	0.01931	0.00749
	DREAM-S3	0.01430	0.00486	0.02000	0.00826
DREAM		0.01639	0.00628	0.02285	0.00870

5 CONCLUSION

In this paper, we propose DREAM for social RS. DREAM tries to model both usersâĂŹ dynamic interests and their friendsâĂŹ temporal influences. Specifically, in each session, to solve the sparsity of social relations, we design a GloVe-based method to increase the number of friends, and utilize relational-GAT to integrate influences from friends. And then we build TIE modules to encode the outputs form historical sessions by recursively combining the features encoded by relational-GAT modules and that from last TIE module. By doing so, the user representations involve both the user's dynamic interests and the dynamic influence from her friends. In the extensive experiments on the public datasets, DREAM significantly outperforms the state-of-the-art solutions.

REFERENCES

[1] W. Fan, Y. Ma, Q. Li, Y. He, Y. E. Zhao, J. Tang, and D. Yin. 2019. Graph Neural Networks for Social Recommendation. In *WWW*. 417–426.

[2] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk. 2016. Session-based Recommendations with Recurrent Neural Networks. In *ICLR*.

[3] K. Kang, J. Park, W. Kim, H. Choe, and J. Choo. 2019. Recommender System Using Sequential and Global Preference via Attention Mechanism and Topic Modeling. In *CIKM 2019*. 1543–1552.

[4] W. Kang and J. McAuley J. 2018. Self-Attentive Sequential Recommendation. In *ICDM*. 197–206.

[5] D. P. Kingma and J. Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*, Y. Bengio and Y. LeCun (Eds.).

[6] J. Pennington, R. Socher, and C. D. Manning. 2014. Glove: Global Vectors for Word Representation. In *EMNLP*. 1532–1543.

[7] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *UAI*. 452–461.

[8] W. Song, Z. Xiao, Y. Wang, L. Charlin, M. Zhang, and J. Tang. 2019. Session-Based Social Recommendation via Dynamic Graph Attention Networks. In *WSDM*.

[9] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio. 2018. Graph Attention Networks. In *ICLR*.

[10] F. Xu, J. Lian, Z. Han, Y. Li, Y. Xu, and X. Xie. 2019. Relation-Aware Graph Convolutional Networks for Agent-Initiated Social E-Commerce Recommendation. In *CIKM*. 529–538.

[11] T. Zhao, J. J. McAuley, and I. King. 2014. Leveraging Social Connections to Improve Personalized Ranking for Collaborative Filtering. In *CIKM*. 261–270.