Lab Seminar

GNN for Social Recommender Systems

(DGRec & DREAM)

HTET ARKAR
School of Computer Science and Engineering
Chung-Ang University

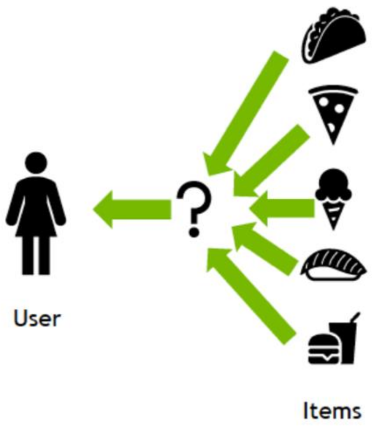
2024-09-27

Content

- **Introduction**
- **DGRec & DREAM**
  - Overview
  - Problem & Motivation
  - Proposed Method
  - Experimental Results
  - Conclusion

Introduction

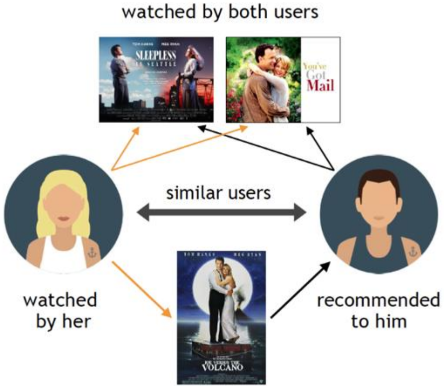
- **Recommendation System (RS)**
  - A machine learning system that predicts how a user would rate an item
  - And ranks/returns those items accordingly
  - Important in the increasingly overloaded age of digital economy



Introduction (Cont'd)

Types of RS

- 1. Collaborative Filtering (CF)
  - Helps predict what users will like based on **patterns of similar users**
  - Subtypes: Memory-based CF and Mode-based CF

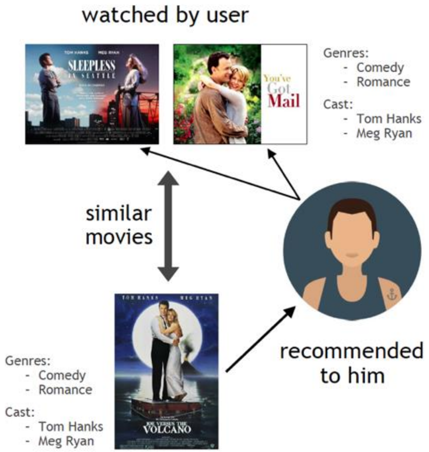


  - Advantages
    - Capable of accurately recommending complex items

Introduction (Cont'd)

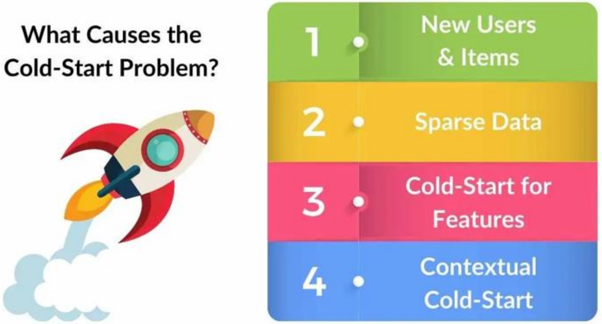
Types of RS

- 2. Content-Based Filtering
  - Based on a description of the item and a profile of the user's preferences
  - Treats as a user-specific classificaiton provlem
  - Learns a classifier for the user's likes and dislikes



Introduction (Cont'd)

- General Problems
  - **Data Sparsity**
    - **Insufficient** data available to accurately predict
  - **Cold start problem**
    - **Rely heavily on past user interactions/item features**
    - Not enough data to make accurate recommendations for a **new user or item**



Introduction (Cont'd)

- General Problems
  - **Scalability**
    - A large amount of computation power is often necessary to calculate
  - **Limited Discovery:**
    - Prevents users from discovering new and potential interesting items
  - **Feature Engineering Dependency**:
    - Generates poor/irrelevant recommendations if feature set is incomplete/lacks depth

Introduction (Cont'd)

□ Other Problems

- RSs tend to utilize all historical user-item interactions
- Learning each user's long-term and static preferences on items

"All of the historical interactions of a user are equally important to her current preferences."

BUT not be the reality…!

Shoujin Wang, Longbing Cao, Yan Wang, Quan Z. Sheng, Mehmet A. Orgun, and Defu Lian. 2021.
A survey on session-based recommender systems. *ACM Computing Surveys* 54,7(2021),1–38.

Introduction (Cont'd)

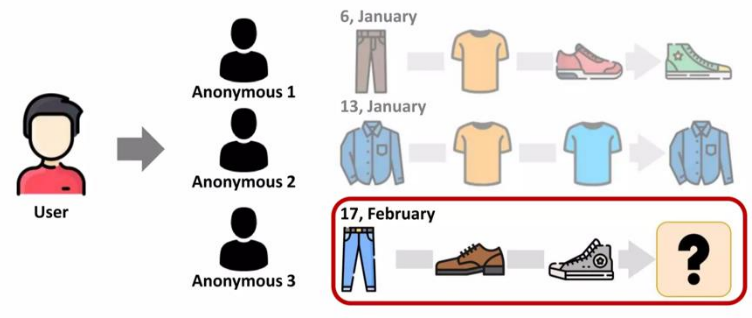
- **Reasons**
  - User's choice depends on
    - Long-term historical and **short-term recent preferences**
    - Time-sensitive context
  - User's preference towards to be **dynamic**
    - i.e. evolving over time

"Session-based Recommender Systems (*SBRSs*)"

Introduction (Cont'd)

Key Features of SBRSs

- Sessions as the basic input unit
  - SBRSs use each session of interactions to learn user preferences
- Captures Both:
  - **User's Short-term Preferences**: Derived from recent sessions
  - **Preference Dynamics**: Reflecting changes in user preferences across different sessions
- Outcome: **More accurate and Timely Recommendations**



Content

Paper-1

Session-Based Social Recommendation via Dynamic Graph Attention Networks

(DGRec)

Weiping Song [1], Zhiping Xiao [2], Yifan Wang [1], Laurent Charlin [3], Ming Zhang [1], Jian Tang [3]

[1] Peking University, [2] UC Berkeley, [3] Mila & HEC Montreal

ACM International Conference on Web Search and Data Mining, 2019

Overview

❖ Background

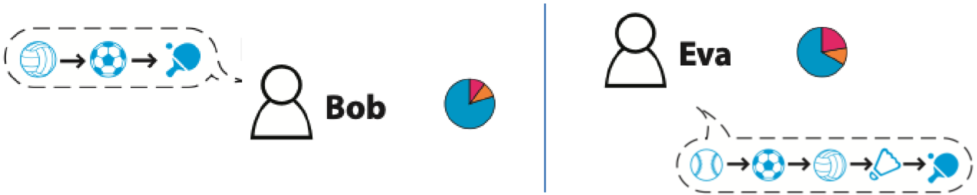
- Online social communities(e.g. Facebook, Twitter) are hugely popular and important in users' daily lives
- Users create, share, and consume information on these platforms
- Recommender systems are critical for these platforms to:
  - ☐ Surface relevant content to users
  - ☐ Improve long-term user engagement

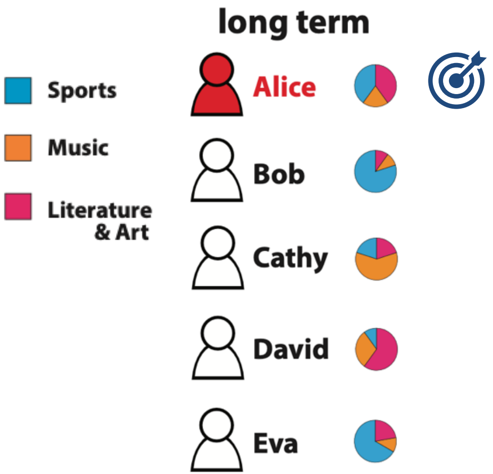
Motivation Example

❖ Scenario

- User Alice has different interests in two sessions: **sports** and **literature**

- **Influences**: (in sports)
  - ☐ Friends Bob and Eva: Sport fans (**long-term**)
  - ☐ Browsing: Sport items (**short-term**)



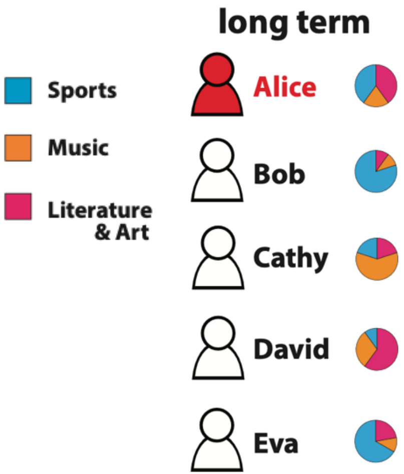


Motivation Example (Cont'd)

❖ Scenario

- User Alice has different interests in two sessions: **sports** and **literature**

- **Influences**: (in literature)
  - Friend David: Interested in literature & arts (**long-term**)
  - Hobby: Activities (**short-term**)



Problem

❖ Challenges

- User interests are **dynamic** and change over time
- Users are **influenced** by their friends' activities
- Influencers can be **context-dependent** (Different friends for different topics)

❖ Key Aspects to Model

- Dynamic user interests
- Context-dependent social influences

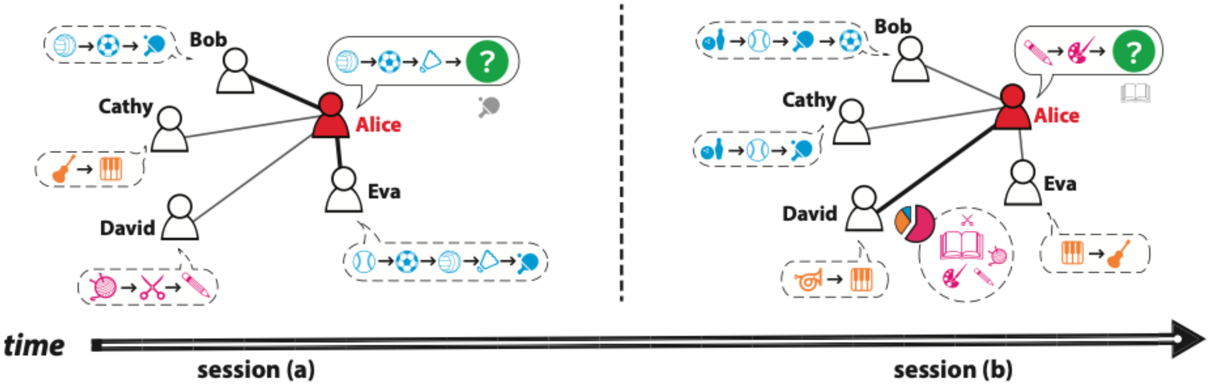
Problem Definition

❖ Session-based Recommendation

- Given a sequence of user behaviors in a session, predict the next item

❖ Session-based Social Recommendation

- Session-based interests + Context-dependent social influence from friends



Proposed Method

❖ Model Components

- **Recurrent Neural Networks (RNN)**
  - ☐ To model users' session-based interests
- **Dynamic Graph Attention Network**
  - ☐ To dynamically model social influences based on the context
- **Friend Influence**
  - ☐ Friends' short-term and long-term preferences are combined using attention

❖ Core Idea

- Dynamic graph structure changes over time based on user's current interests and friends' influence

Architecture

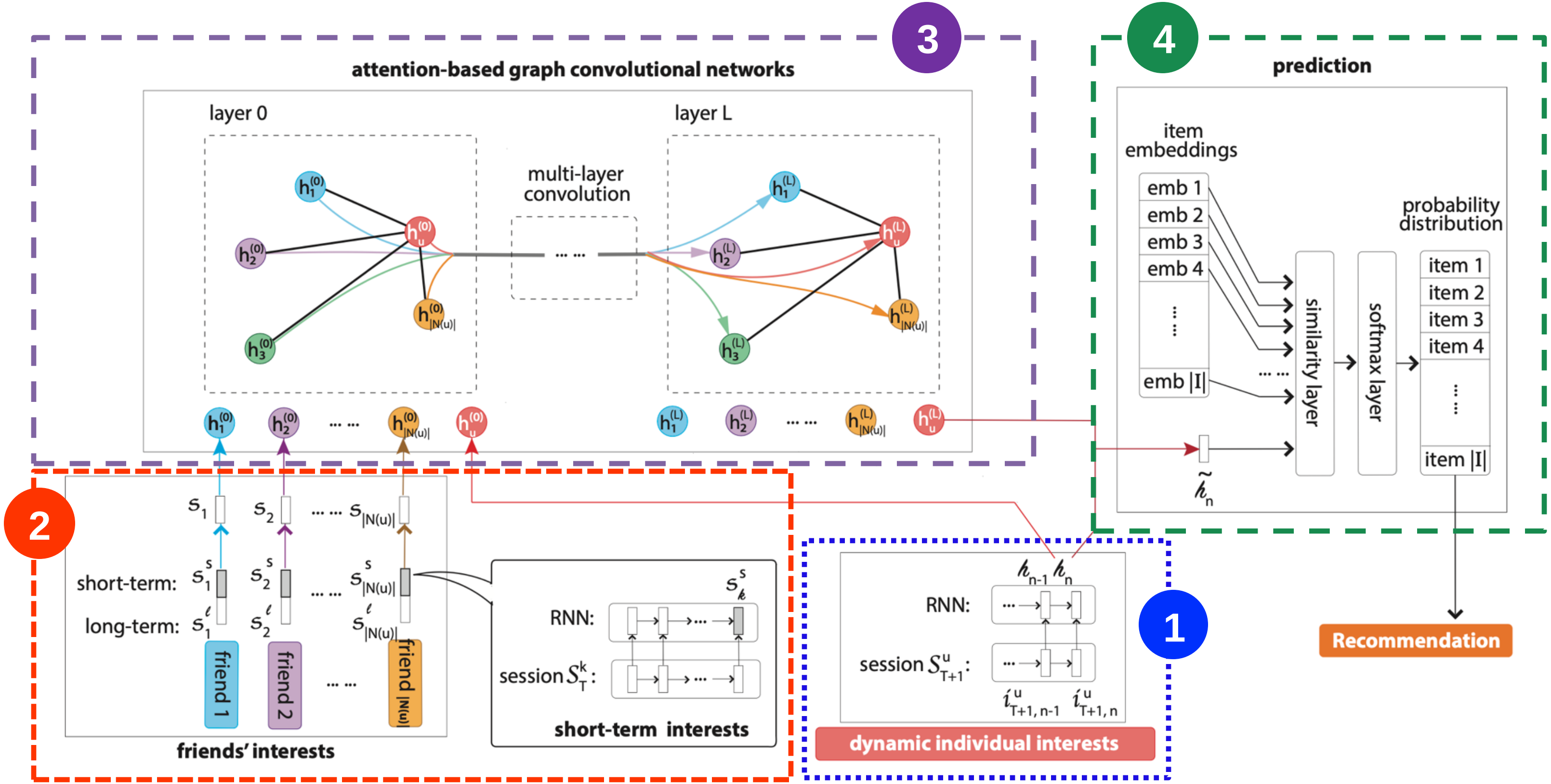


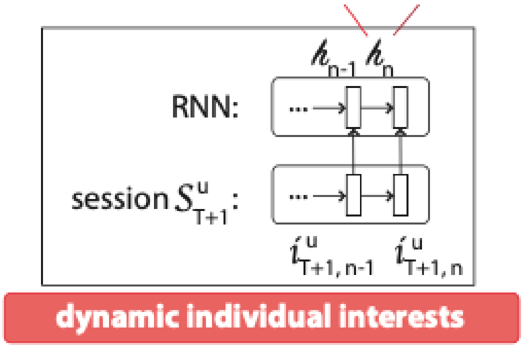
Figure 2: A schematic view of our proposed model for dynamic social recommendation.

Architecture (Cont'd)

❖ (1) Dynamic Individual Interests

- Modeled via RNN
- Capture the sequence of items a user interacts with in a session

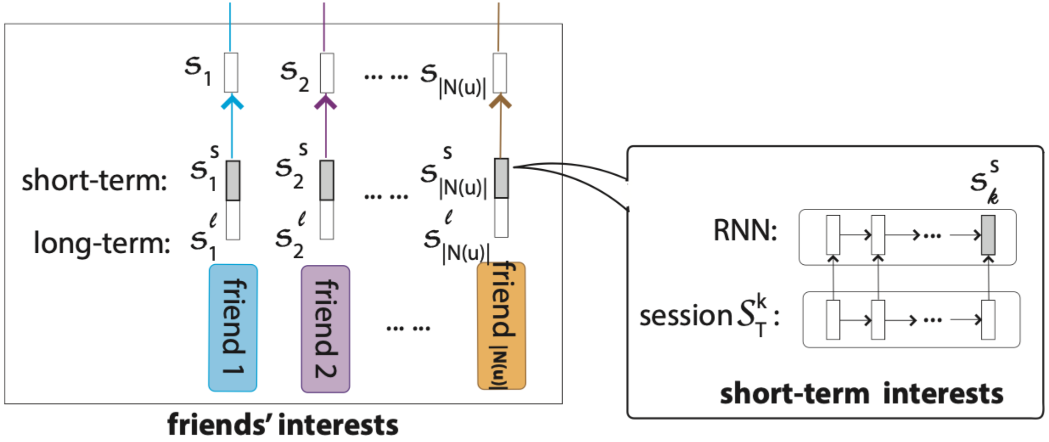
$$h_n = f(i^u_{T+1,n}, h_{n-1}),$$



Architecture (Cont'd)

❖ (2) Dynamic Social Interests:

- Represent friends' short- and long-term preferences
  - ☐ Short-term : current session
  - ☐ Long-term : historical average



Architecture (Cont'd)

❖ (2) Dynamic Social Interests

- Short-term preference
  - ☐ Represents a friend's **most recent activities**
    - (e.g., their latest session of consumed items)
  - ☐ These are **dynamic** and **session-specific**,
    - Reflecting what friends have interacted with recently

- Approach:
  - ☐ Uses an RNN to model the sequence of items a friend consumed in last session

$$s_k^{short} = h_{last}$$

  - ☐ h_{last} : final hidden state of RNN

Architecture (Cont'd)

❖ (2) Dynamic Social Interests

- Long-term preference
  - Represents a friend's overall preferences accumulated over time
  - These are static, reflecting **general, long-term** behavior patterns
    - That don't change frequently

- Approach:
  - Uses a learned embedding vector
  - Capturing a friend's overall preferences across multiple sessions

$$s_k^{long} = \mathbf{W}_u[k, :]$$

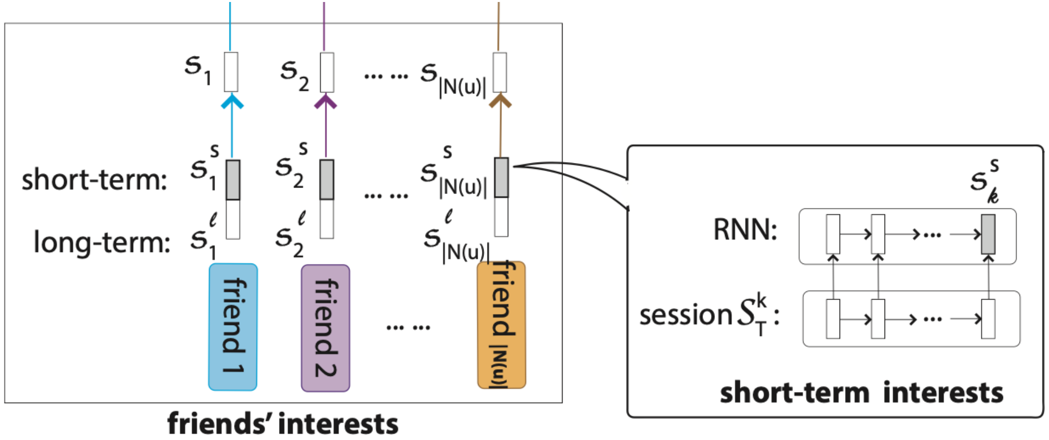
  - k_{th} row of the user embedding matrix W_u

Architecture (Cont'd)

❖ (2) Dynamic Social Interests

- Friend node's feature
  - ☐ Concatenation of the friend's short-term and long-term preferences

$$s_k = ReLU(\mathbf{W}_1[s_k^s; s_k^l]),$$



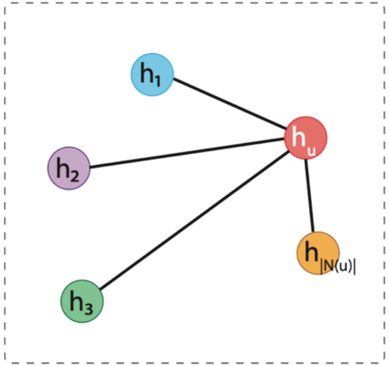
Architecture (Cont'd)

❖ (3) Dynamic Graph Attention Network

- Context-dependent Social Influences
- Constructing **Dynamic Feature Graph**
  - ☐ The graph for a user $\boldsymbol{u}$ consists of $|\boldsymbol{N(u)}|+1$ nodes, where $\boldsymbol{N(u)}$ is the set of user u's friends
  - ☐ **Edges** represent the social connections between the target user and each friend
- Target user node features:

$$h_u^{(0)} = h_{current}$$

$h_{current}$: user's interests in the current session



- Friends' Node Features:

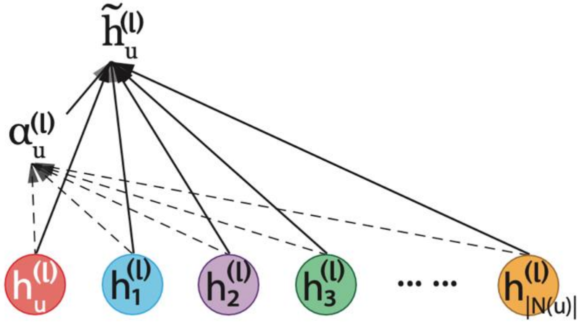
$$h_k^{(0)} = s_k$$

s_k : concatenation of the friend's short-term and long-term preferences

Architecture (Cont'd)

❖ (3) Dynamic Graph Attention Network

- Graph Attention Network (GAT)
  - ☐ To determine how much influence each friend should have on the target user
  - ☐ Attention Mechanism: $$\alpha_{uk}^{(l)} = \frac{exp(f(h_u^{(l)}, h_k^{(l)}))}{\sum_{j \in N(u) \cup \{u\}} exp(f(h_u^{(l)}, h_j^{(l)}))},$$ (level of influence)
  - ☐ Massage Passing:
    - Between the target user and friends based on the computed attention weights
  - ☐ Feature Aggregation: $$\tilde{h}_u^{(l)} = \sum_{k \in N(u) \cup \{u\}} \alpha_{uk}^{(l)} h_k^{(l)},$$
    - Aggregates the information from all friends using a weighted sum

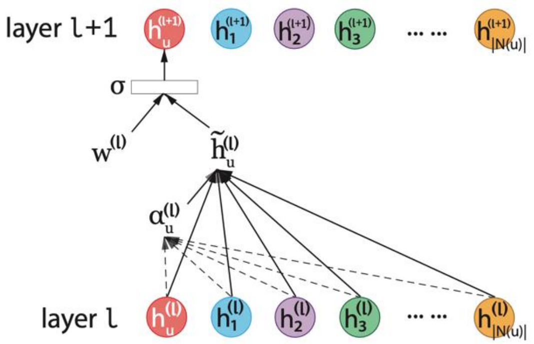


(3) Dynamic Graph Attention Network

❖ Attention-based Graph Convolutional Networks

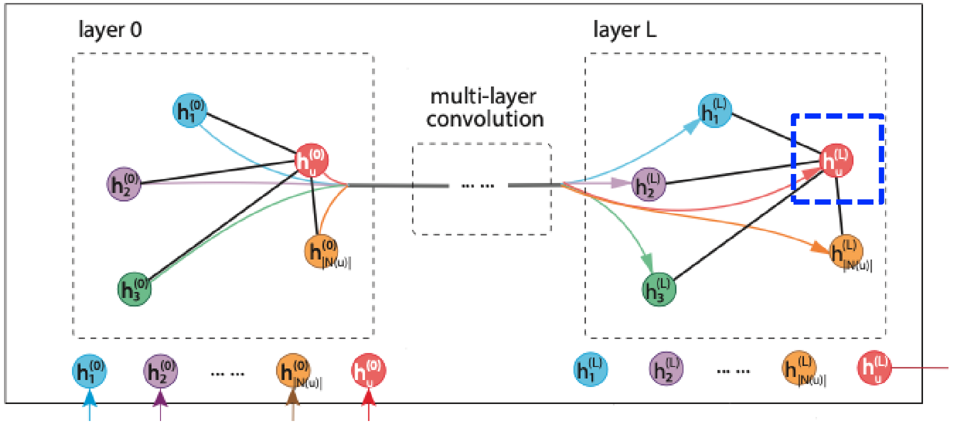
- Stacking the attention layer L times
- $h_u^{(L)}$: The combined (social-influenced) representation

$$h_u^{(l+1)} = ReLU(\mathbf{W}^{(l)}\tilde{h}_u^{(l)})$$



=

attention-based graph convolutional networks



Architecture (Cont'd)

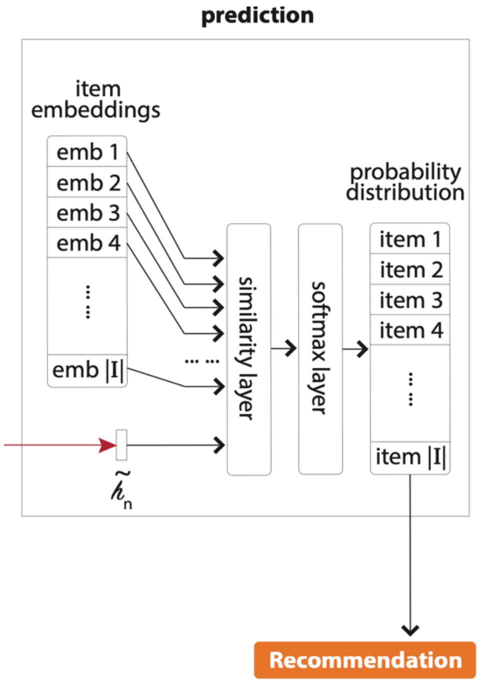
❖ (4) Recommendation

- Final User Representation
  - ☐ Combining user's dynamic interests and social influence

$$\hat{h}_n = \mathbf{W}_2[h_n; h_u^{(L)}],$$

- Next-item Prediction
  - ☐ Probability that next item will be, y
  - ☐ Using a softmax function

$$p(y|i^u_{T+1,1}, \ldots, i^u_{T+1,n}; \{\vec{S}^k_T, k \in N(u)\}) = \frac{\exp(\hat{h}_n^\top z_y)}{\sum_{j=1}^{|I|} \exp(\hat{h}_n^\top z_j)},$$



Architecture (Cont'd)

❖ Training

- Using log-likelihood function
  - Measures how likely the model is to predict the actual items users interacted with during their sessions
  - To maximize :

$$\sum_{u \in U} \sum_{t=2}^{T} \sum_{n=1}^{N_{u,t}-1} \log p(i^u_{t,n+1} | i^u_{t,1}, \ldots, i^u_{t,n}; \{\vec{S}^k_{t-1}, k \in N(u)\}).$$

  - Explanation:
    - U: Set of all users
    - T: Number of sessions
    - $N_{u,t}$: Number of items in session t for user u.
    - $i_{u,t,n+1}$: The actual item the user interacted with next.

Experiment

❖ Quantitative Results

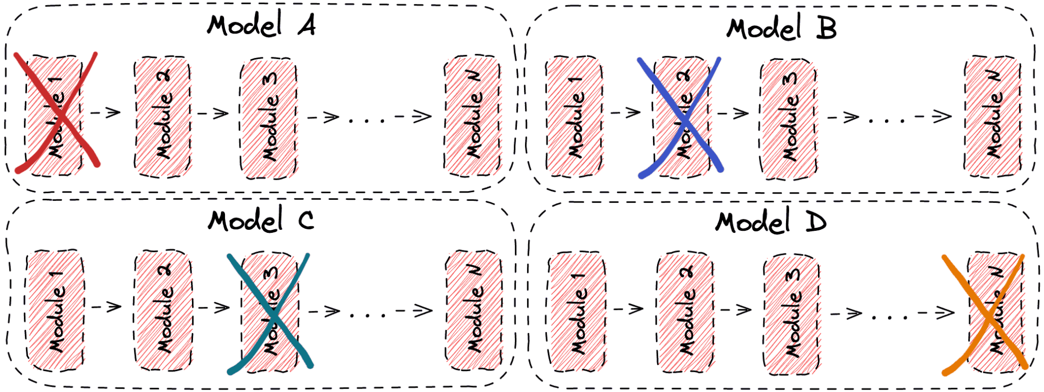
- Showing the **necessity** of modeling context-dependent social influences

Model Class	Model	Douban		Delicious		Yelp	
		Recall@20	NDCG	Recall@20	NDCG	Recall@20	NDCG
Classical	ItemKNN [22]	0.1431	0.1635	0.2729	0.2241	0.0441	0.0989
	BPR-MF [27]	0.0163	0.1110	0.2775	0.2293	0.0365	0.1190
Social (Content-independent)	SoReg [24]	0.0177	0.1113	0.2703	0.2271	0.0398	0.1218
	SBPR [41]	0.0171	0.1059	0.2948	0.2391	0.0417	0.1207
	TranSIV [38]	0.0173	0.1102	0.2588	0.2158	0.0420	0.1187
Temporal (Dynamic User Interest)	RNN-Session [13]	0.1643	0.1854	0.3445	0.2581	0.0756	0.1378
	NARM [21]	0.1755	0.1872	0.3776	0.2768	0.0765	0.1380
Social + Temporal (Ours)	DGRec	**0.1861**	**0.1950**	**0.4066**	**0.2944**	**0.0842**	**0.1427**

Experiment

❖ Ablation Study

“Machine learning system의 building blocks을 제거해서
전체 성능에 미치는 효과에 대한 insight를 얻기 위한 과학적 실험”



Ablation Study

❖ Self v.s. Social

- DGREC_{self} > DGREC_{social} : overall users' individual interests have higher impact
- DGREC : crucial to model both a user's current interest + dynamic social influences

Data Sets	Models	Recall@20	NDCG
Douban	DGREC_{self}	0.1643	0.1854
	DGREC_{social}	0.1185	0.1591
	DGREC	0.1861	0.1950
Delicious	DGREC_{self}	0.3445	0.2581
	DGREC_{social}	0.3306	0.2516
	DGREC	0.4066	0.2944
Yelp	DGREC_{self}	0.0756	0.1378
	DGREC_{social}	0.0690	0.1356
	DGREC	0.0842	0.1427

- DGREC_{self} : a model of the user's current session only
- DGREC_{social} : a model using context-dependent social influence features only

Variation of DGRec

❖ Number of Convolutional Layers

- Determines the depth of social influence
- The more layers allows influence from higher-order friends (friends of friends, etc.)

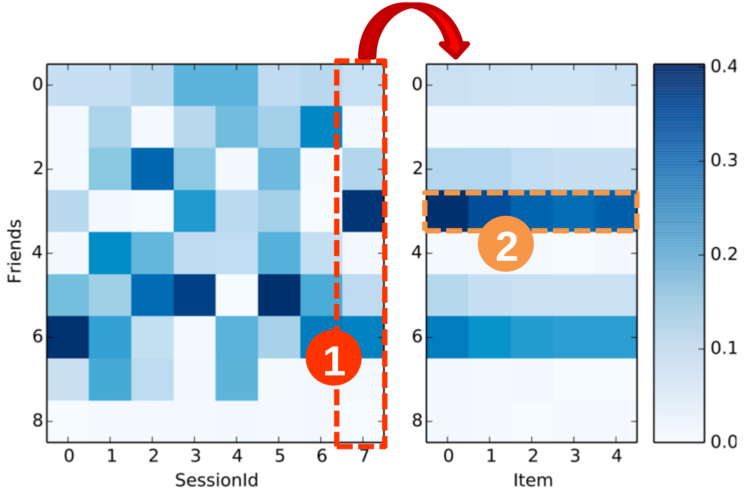
Data Sets	Conv. Layers	Recall@20	NDCG
Douban	1	0.1726	0.1886
	2	0.1861	0.1950
	3	0.1793	0.1894
Delicious	1	0.4017	0.2883
	2	0.4066	0.2944
	3	0.4037	0.2932
Yelp	1	0.0760	0.1387
	2	0.0842	0.1427
	3	0.0846	0.1423

- **Over-fitting or noises introduced**
- **Two layers are enough**

Variation of DGRec (Cont'd)

❖ Exploring Attention

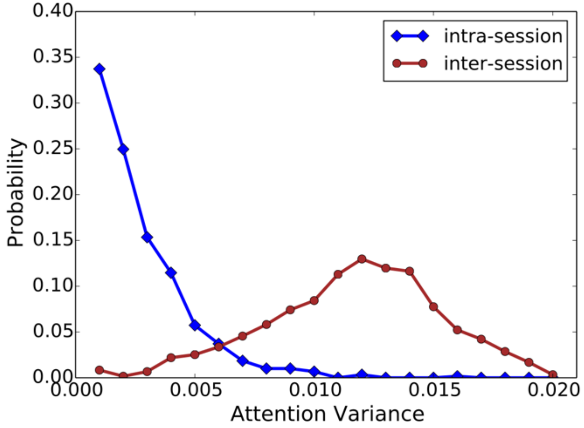
- (1) User allocates her attention to different friends across different sessions
- (2) Distribution of attention is relatively stable within a single session



Variation of DGRec (Cont'd)

❖ Exploring Attention

- **Intra-session** –> Lower : Users' interests tend to be focused within a short time
- **Inter-session** –> Higher : User is more likely to trust different friends in different sessions



Conclusion

- **Graph Convolutional Networks (GCNs)** for session-based social recommendation
- User representations captures **current interest**
- **Friends' Influence** is aggregated using attention-based GCNs
- Item recommendations is from combined user and social preferences

Content

Paper-2

A Dynamic Relational-Aware Model for Social Recommendation

(DREAM)

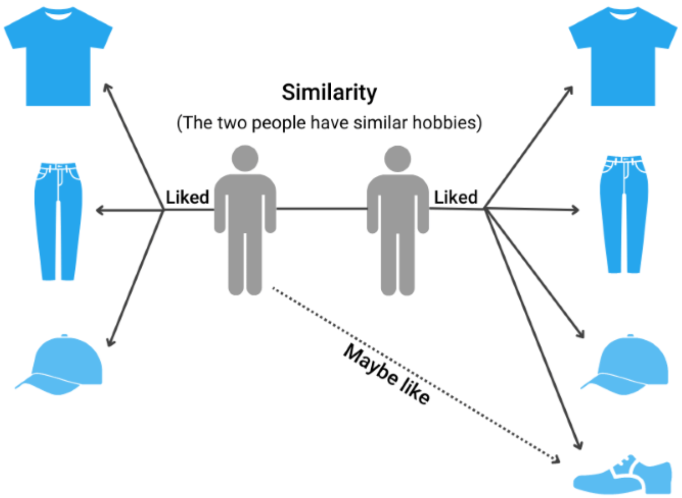
Liqiang Song, Ye Bi, Mengqiu Yao, Zhenyu Wu, Jianming Wang, Jing Xiao

Ping An Technology Shenzhen Co., Ltd

ACM International Conference on Information and Knowledge Management (CIKM ’20)

Introduction

- **Social Recommender System (SocialRS)**
  - Social connections are used to enhance the performance of RS
  - Users are influenced by friends’ preferences and behaviors



Problems

- **Challenges in SocialRS**
  - **Dynamic user preferences** and **friends' influences** are continuously evolving
  - * Social relations are often **sparse** *
    - Making the system difficult to generate meaningful recommendations

- **DGREC (in previous work)**
  - Dynamic user behavior with RNN
  - Social Influence with GAT

  - **Problems**
    - Ignoring the effects of temporal dependency among different sessions
    - Relying only on social friends is far from satisfactory

Solution

Dynamic RElational-Aware Model

A unified model to address dynamic user interests

and sparse social relations

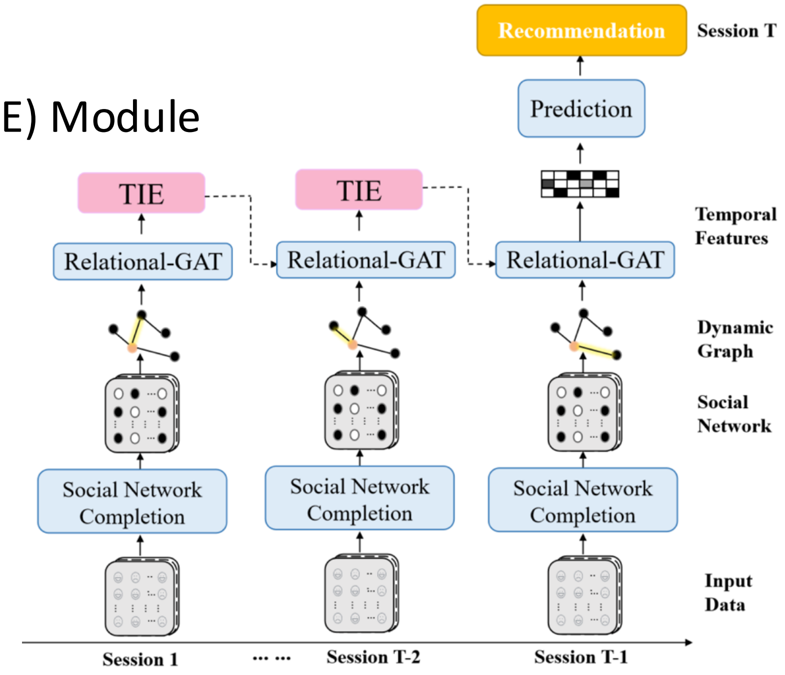
using virtual friends and temporal information encoding (TIE) modules

Proposed Method - DREAM

- **Key Contributions:**
  - Models both **users' dynamic interests** and their friends' **temporal influences**
  - **Virtual friends:** To alleviate the sparsity of social relations
  - **relational-GAT**: To integrate real and virtual friends’ influences
  - **Temporal information encoding (TIE)**: To update user representations across sessions

Proposed Method - DREAM Framework

- There are the following four parts
  - (1) Social Network Completion
  - (2) Relational-GAT Module
  - (3) Temporal information encoding (TIE) Module
  - (4) Prediction

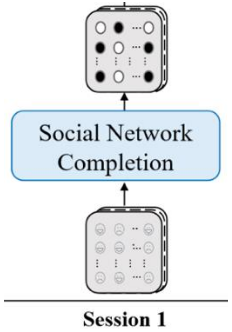


DREAM Framework

(1) Social Network Completion

- Virtual Friends
  - Defined as users with **similar** consumption habits
  - Connection strength is based on **similarity** calculated using a GloVe-based mechanism

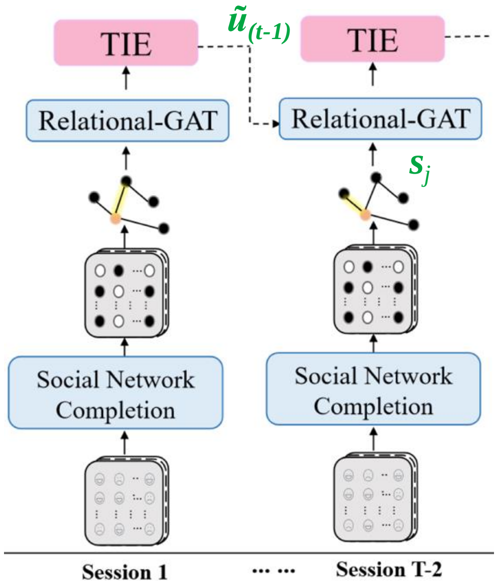
$$s^V_{p,q} = \mathrm{softmax}(\langle \boldsymbol{g}_{u_p}, \boldsymbol{g}_{u_q} \rangle) = \frac{\exp(\langle \boldsymbol{g}_{u_p}, \boldsymbol{g}_{u_q} \rangle)}{\sum_{u_l, u_s \in \mathcal{U}} \exp(\langle \boldsymbol{g}_{u_l}, \boldsymbol{g}_{u_s} \rangle)}$$



- How it Works:
  - Similarity among users is computed
  - Top-k similar users are selected as virtual friends
  - Virtual friends are added to the user's social network to complete the graph

DREAM Framework

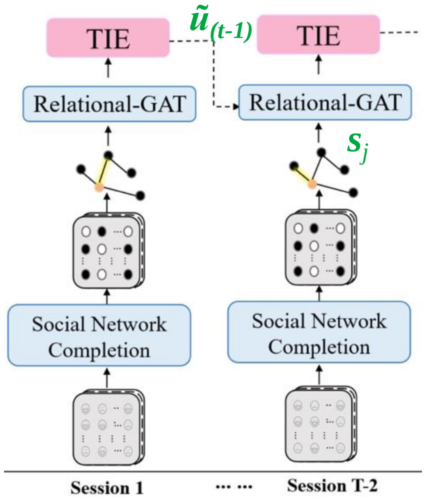
- **Node Representation**
  - (1) Friends' node representations :
    - $\tilde{\boldsymbol{u}}_{(t\text{-}1)}$: Calculated at (t-1)-th session
  - (2) Users' short-term interests :
    - $s_j = \mathrm{GRU}(S^j_{t\text{-}1})$



DREAM Framework

(2) Relational-GAT Module

- Captures the different types of relationships between users (real vs virtual)
- Input
  - For target user u: $h_u(0) = \tilde{u}_{(t\text{-}1)}$ (from previous TIE)
  - For friends: $h_j(0) = s_j$ (short-term interest from GRU)
- Attention score: $$\alpha_{uk} = \frac{\exp(f_r(\boldsymbol{h}_u^{(0)}, \boldsymbol{P}_r \boldsymbol{h}_k^{(0)}))}{\sum_{u_j \in \mathcal{N}(u)\cup\{u\}} \exp(f_r(\boldsymbol{h}_u^{(0)}, \boldsymbol{P}_r \boldsymbol{h}_j^{(0)}))}, \quad \forall w \in \mathcal{N}(u).$$
- Information aggregation: $$\boldsymbol{h}_u = \sigma\left(\sum_{u_j \in \mathcal{N}(u)\cup\{u\}} \alpha_{uj} \boldsymbol{h}_j^{(0)}\right),$$
- Output : user's final representation, $u_t = h_u$



DREAM Framework

(3) Temporal information encoding (TIE) Module

- Combines the Relational-GAT encoded features with historical session data
  - To capture evolving user preferences
- A GRU-like mechanism is used
  - To model long-term preferences
  - To account for how friends' influences change over time
- Encoding Procedure:

$$
\begin{aligned}
\boldsymbol{u}_q &= \boldsymbol{W}_q^t \widetilde{\boldsymbol{u}}_{t-1} + \boldsymbol{b}_q^t \\
\boldsymbol{u}_e &= \boldsymbol{W}_e^t \boldsymbol{u}_t + \boldsymbol{b}_e^t \\
\widetilde{\boldsymbol{h}}_t &= \tanh(\boldsymbol{W}_h^t \boldsymbol{u}_t + \boldsymbol{u}_e \circ \boldsymbol{U}_h^t \widetilde{\boldsymbol{u}}_{t-1} + \boldsymbol{b}_h^t) \\
\widetilde{\boldsymbol{u}}_t &= (\mathbf{1} - \boldsymbol{u}_q) \circ \widetilde{\boldsymbol{u}}_{t-1} + \boldsymbol{u}_q \circ \widetilde{\boldsymbol{h}}_t,
\end{aligned}
$$

DREAM Framework

Prediction

- Target user representation : final output of realtional-GAT module ($\boldsymbol{u_T}$)
- Target item $\boldsymbol{v}$
- Prediciton calculation: $\hat{y}_{uv} = \sigma(f(\boldsymbol{u_T}, \boldsymbol{v}))$
- Loss funciton: $$\sum_{(u,v)\in R} -(y_{uv} \log \sigma(f(\boldsymbol{u}_T, \boldsymbol{v})) + (1 - y_{uv}) \log(1 - \sigma(f(\boldsymbol{u}_T, \boldsymbol{v}))))$$
  - Where $\boldsymbol{y_{uv}}$ is the ground truth label { 1 : positive interactions
    0 : negative interactions

Experiment

Overall Performance

- [1] Reflect the power of temporal information

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%

Experiment

Overall Performance

- Three reasons that DREAM outperforms
  - (1) Scoial Network using virtual friends
    - To express target user's dynamic and stastic interests
  - (2) Combination of (user historical and current) representation as input of TIE
    - The importance of learning the evolution of target user interests
  - (3) Using multiple temporal session information
    - Reflecting the evolution of target user dynamic interests over time

Experiment

☐ Ablation Study

Model Components		Epinions R@10	Epinions MRR	Movie R@10	Movie 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

- Using **both** real and virtual friends' information gets better performance
- relational-GAT catches the difference between two social information
- Fully varify the **temporal effects** in capturing users' interests
- The more sessions information does not mean better performance

Conclusion

- Model both Users' dynamic interests and friends' temporal influences

- Introduction of virtual friends helps alleviate **data sparsity**

- Capturing and integrating social influences from both real and virtual friends

Thank You!

HTET ARKAR (hak3601@cau.ac.kr)