Double-Scale Self-Supervised Hypergraph Learning for Group Recommendation

Junwei Zhang (Chongqing University), **Min Gao** (Chongqing University),
Junliang Yu (The University of Queensland), **Lei Guo** (Shandong Normal University),
Jundong Li (University of Virginia), **Hongzhi Yin** (The University of Queensland)

jw.zhang@cqu.edu.cn

Backgrounds

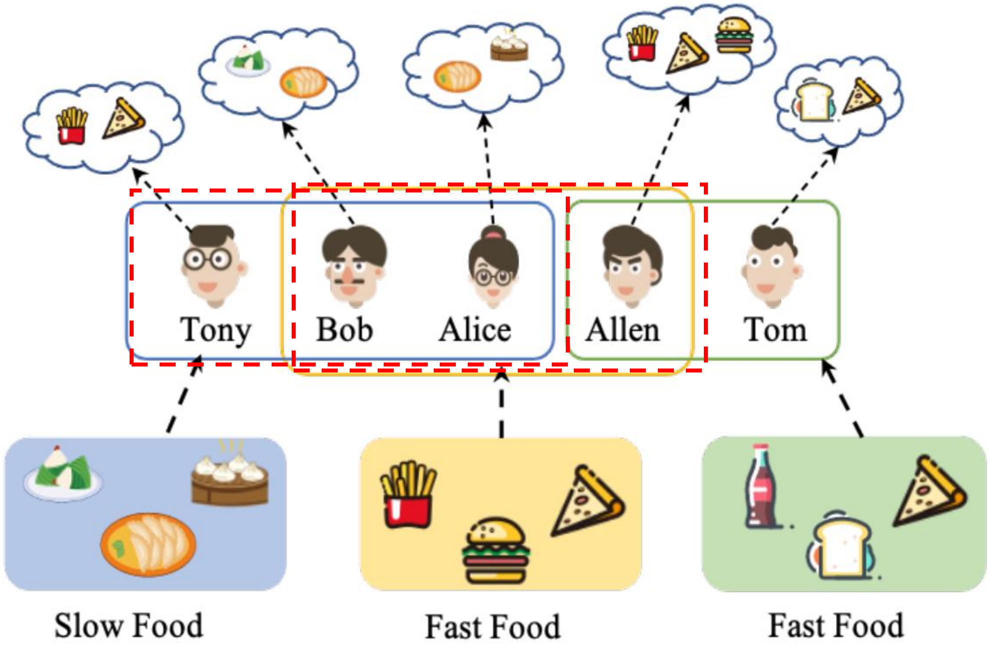
Group Recommendation Task

- Daily Activity
- Answer and Question Websites
- Social Media

……

Challenge and Limitation

- The collective decisions tend to be **dynamic.**
- The group and user relations are **complicated and high-order.**
- The group interaction data is **sparsity.**

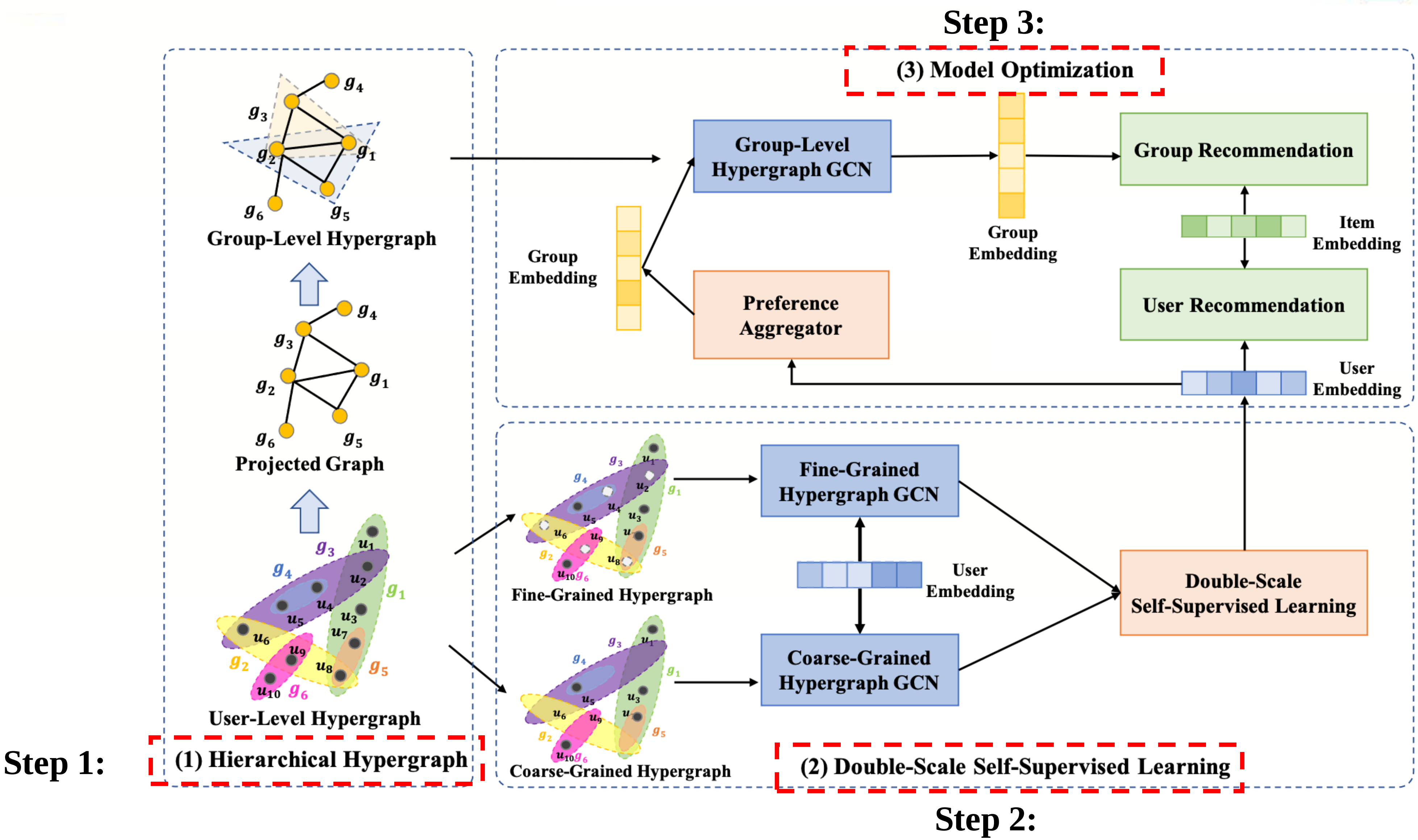


Motivation

- **Utilizing Hierarchical Hypergraph to model high-order user member relations**
  - To capture the **intra-group interactions**, we adopt the hypergraph convolutional network to aggregate the related users and generate the dynamic user embeddings.
  - To capture the **inter-group interactions**, we adopt the triangular motifs to select the most relevant groups information.

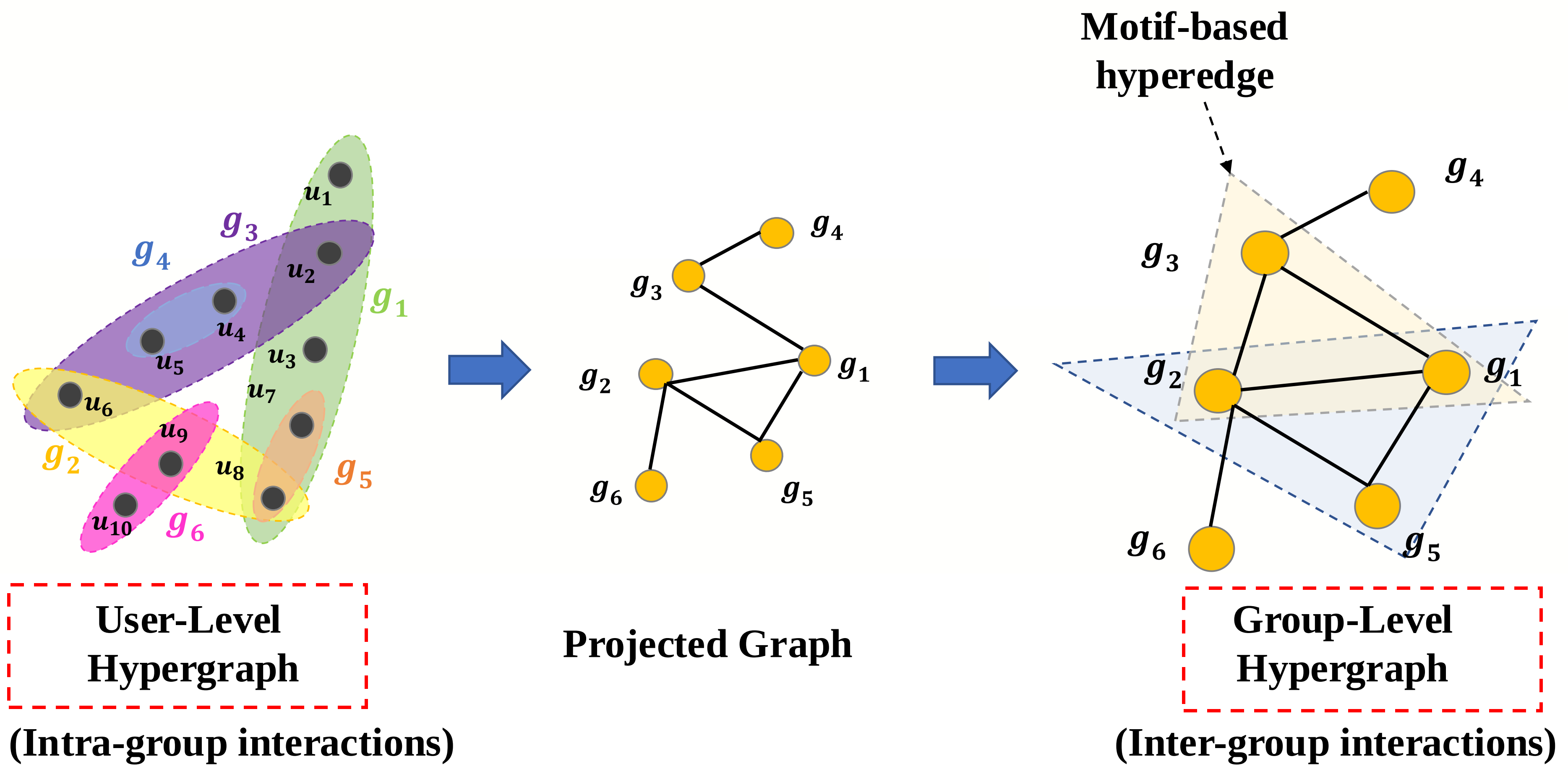
- **Utilizing the self-supervised learning to alleviate data sparsity problem**
  - We design two scale-grained node dropping strategy as the self-supervision signals from the raw data to help the training of the hierarchical hypergraph convolutional network.

Method: Overview



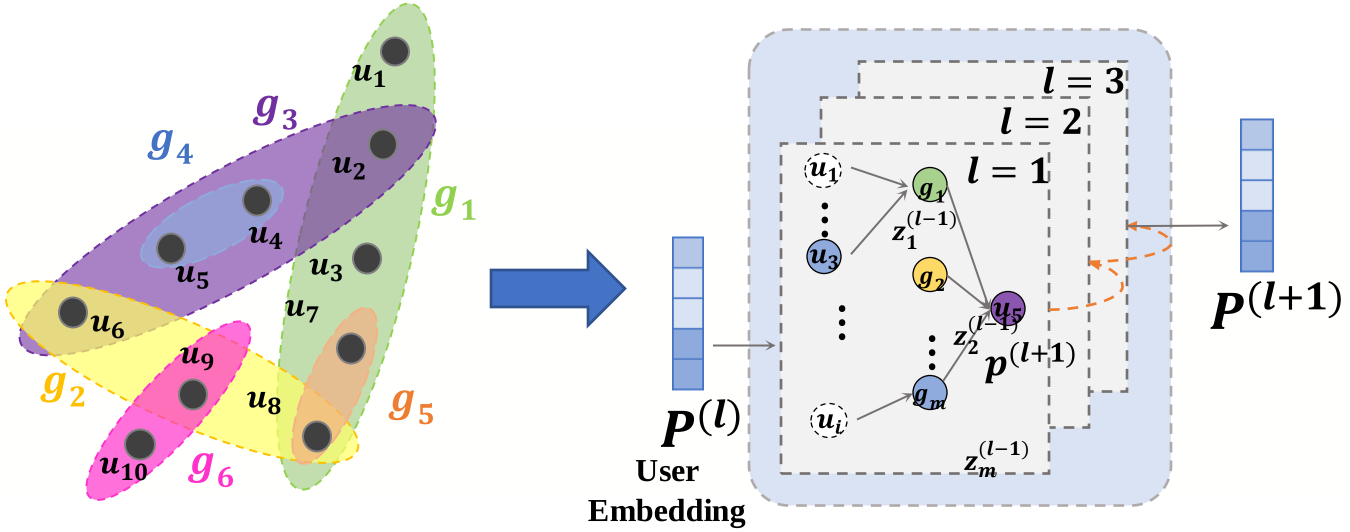
SEPT 1: Hierarchical Hypergraph

- **Hypergraph Construction**



SEPT 1: Hierarchical Hypergraph

- **User-Level Hypergraph Representation**



User-Level Hypergraph G_{ul}

HGNN

$$P^{(l+1)} = \sigma(D_{ul}^{-1} H_{ul} W_{ul} B_{ul}^{-1} H_{ul}^{T} P^{(l)} \Theta^{(l)}), \quad (1)$$

vertex degree matrix

incidence matrix of the user-level hypergraph

hyperedge degree matrix

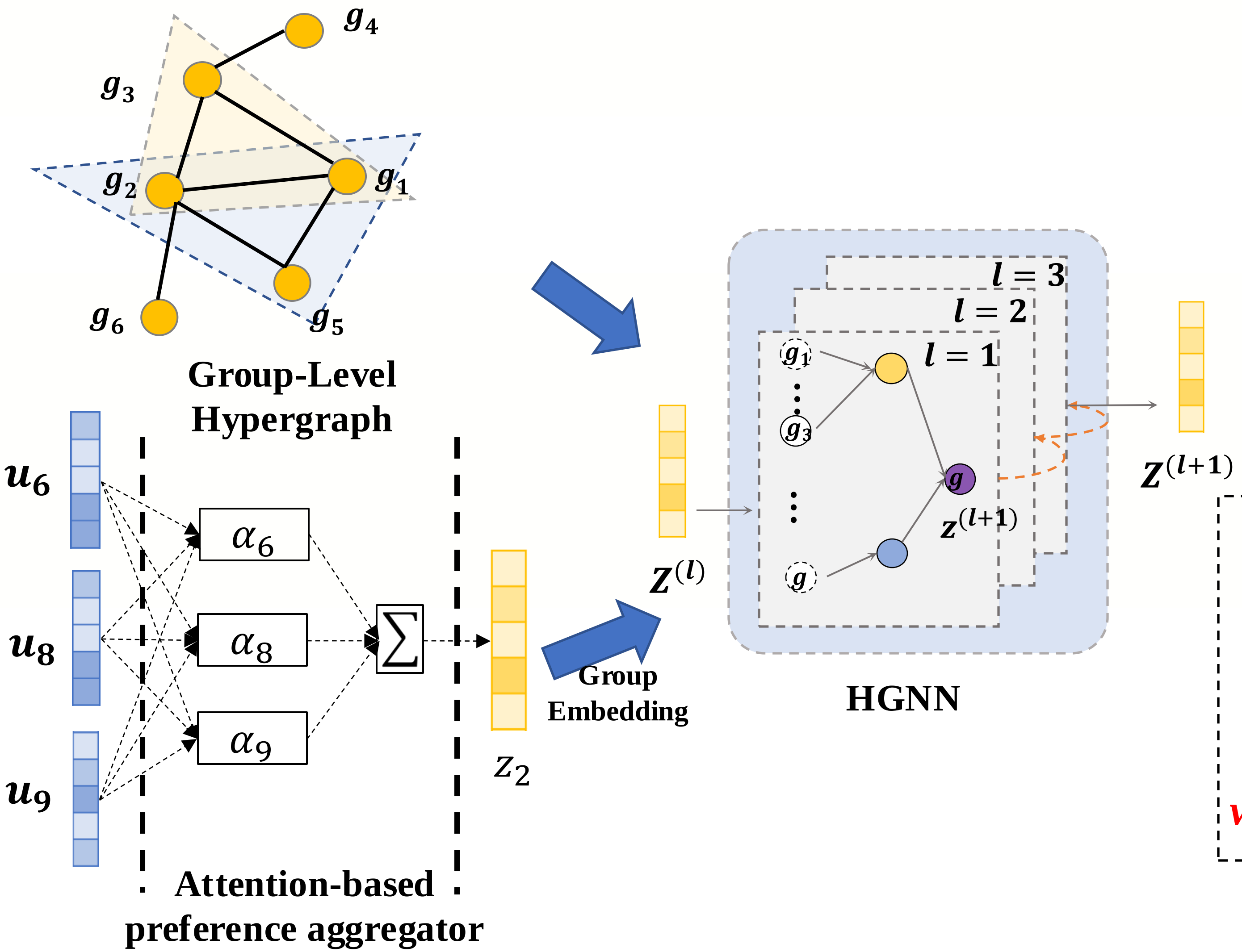
the parameter matrix

To reduce the complexity of the model, we remove the nonlinear activation:

$$P^{(l+1)} = D_{ul}^{-1} H_{ul} B_{ul}^{-1} H_{ul}^{T} P^{(l)} \Theta^{(l)}. \quad (2)$$

SEPT 1: Hierarchical Hypergraph

- **Group-Level Hypergraph Representation**



$$H_{gl} H_{gl}^T = (CC) \odot C, \tag{5}$$

$$Z^{(l+1)} = D_{gl}^{-1} H_{gl} H_{gl}^T Z^{(l)} \Psi^{(l)}, \tag{6}$$

$\boldsymbol{H_{gl}}$ *denotes the* ***motif incidence matrix***

$\boldsymbol{C}$ *is the symmetric adjacency matrix of the projected graph*

$\tilde{\boldsymbol{z}}$ *is the input of the group-level hypergraph*

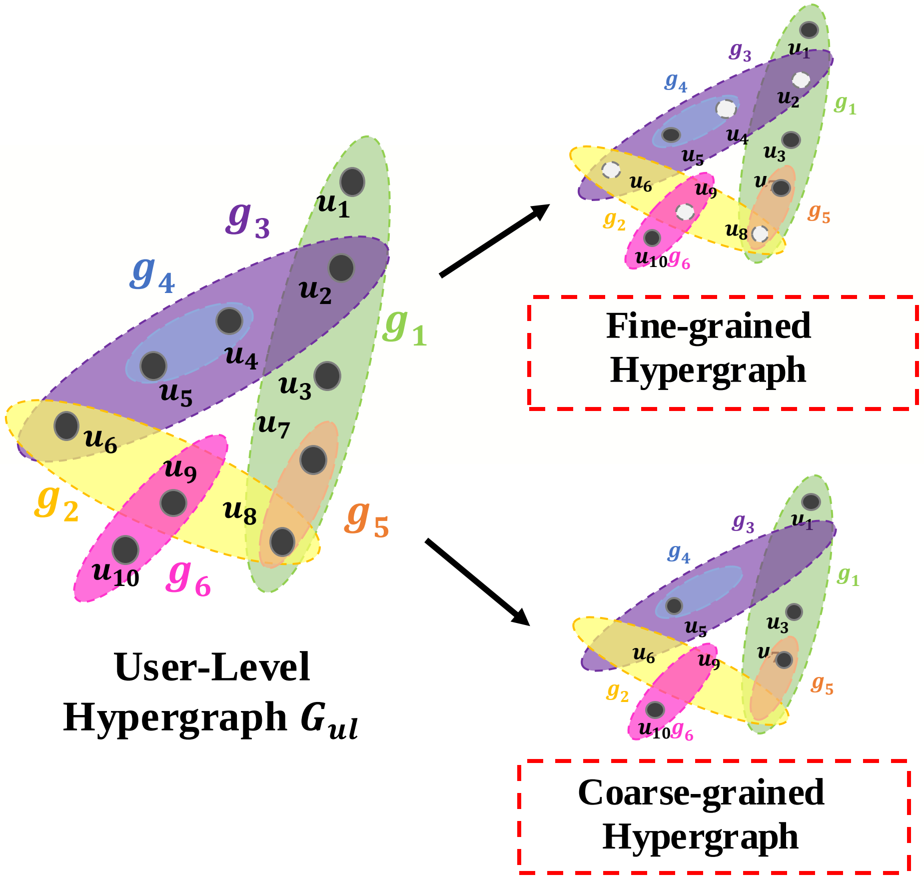
$$\tilde{z}_g = \sum_{u \in g_m} \alpha_u p_u W_{\text{agg}}, \tag{3}$$

α_u *is the* ***weight*** *of the user*

$$\alpha_u = \frac{\exp\left(p_u W_{\text{agg}} x^T\right)}{\sum_{j \in g_m} \exp\left(p_j W_{\text{agg}} x^T\right)}, \tag{4}$$

SEPT 2: Double-Scale Self-Supervised Learning

- **Constructing Double-Scale Grained Self-Supervised Signals**



The column vector of the fine-grained incidence matrix

mask vector

Incidence matrix of the user-level hypergraph

$$h_f = f_{fine}(a_f, h_{ul}) = a_f \odot h_{ul}, \tag{12}$$

$$P''^{(l+1)} = g_f(P''^{(l)}) = D_f^{-1} H_f B_f^{-1} H_f^T P''^{(l)} \Phi^{(l)}, \tag{13}$$

The column vector of the coarse-grained incidence matrix

$$h_c = f_{coarse}(a_c, h_{ul}) = a_c \odot h_{ul}, \tag{10}$$

$$P'^{(l+1)} = g_c(P'^{(l)}) = D_c^{-1} H_c B_c^{-1} H_c^T P'^{(l)} \Gamma^{(l)}, \tag{11}$$

$$P = P' + P''$$

SEPT 2: Double-Scale Self-Supervised Learning

- **Contrastive learning**

$$f_{\mathcal{D}}\left(\boldsymbol{p}_i', \boldsymbol{p}_i''\right) = \sigma\left(\boldsymbol{p}_i' \mathbf{W}_{\mathcal{D}} \boldsymbol{p}_i''^{T}\right), \tag{15}$$

The discriminator function

$$\mathcal{L}_{UU} = -\sum_{i \in U}\left[\log \sigma\left(f_{\mathcal{D}}\left(\boldsymbol{p}_i', \boldsymbol{p}_i''\right)\right) + \sum_{j=1}^{n}\left[\log \sigma\left(1 - f_{\mathcal{D}}\left(\boldsymbol{p}_j', \boldsymbol{p}_i''\right)\right)\right]\right], \tag{14}$$

The user representation of the coarse-grained hypergraph

The user representation of the fine-grained hypergraph

SEPT 3: Optimization

- **HHGR loss function**

$$\hat{r}_{ui} = \boldsymbol{p}_u \tilde{\boldsymbol{q}}_i^T,$$

$$\hat{s}_{gi} = \boldsymbol{z}_g \tilde{\boldsymbol{q}}_i^T,$$

$$\mathcal{L}_{UI} = -\sum_{(u,i,j)\in O} \left(\hat{r}_{ui} - \hat{r}_{uj} - 1\right)^2,$$

$$\mathcal{L}_{GI} = -\sum_{(g,i,j)\in O'} \left(\hat{s}_{gi} - \hat{s}_{gj} - 1\right)^2$$

$$\mathcal{L} = \mathcal{L}_{UI} + \mathcal{L}_{GI}$$

- **S^2 −HHGR loss function**

$$\mathcal{L}_{UI} = -\sum_{(u,i,j)\in O} \left(\hat{r}_{ui} - \hat{r}_{uj} - 1\right)^2,$$

$$\mathcal{L}_{UU} = -\sum_{i\in U}\left[\log\sigma\left(f_{\mathfrak{D}}\left(\boldsymbol{p}_i', \boldsymbol{p}_i''\right)\right) + \sum_{j=1}^{n}\left[\log\sigma\left(1 - f_{\mathfrak{D}}\left(\boldsymbol{p}_j', \boldsymbol{p}_i''\right)\right)\right]\right]$$

$$\mathcal{L}_{GI} = -\sum_{(g,i,j)\in O'} \left(\hat{s}_{gi} - \hat{s}_{gj} - 1\right)^2$$

$$\mathcal{L} = \beta\mathcal{L}_{UU} + \mathcal{L}_{UI} + \mathcal{L}_{GI},$$

Experiments: Settings

- **Research Questions**

RQ1: Compared with the state-of-the-art group recommendation models, how does our model perform?

RQ2: What are the benefits of each component (i.e., the hierarchical hypergraph and the self-supervised learning) in our model?

RQ3: How do the hyper-parameters influence the effectiveness of the S^2-HHGR?

- **Experience Datasets**

Table 1: The statistics of datasets.

Dataset	#User	#Item	#Group	#U-I Feedback	#G-I Feedback
Douban	2,964	39,694	2,630	823,851	463,040
Weeplaces	8,643	25,081	22,733	1,358,458	180,229
CAMRa2011	602	7,710	290	116,344	145,068

Experiments: Recommendation Performance

Table 2: The general recommendation performance comparison on three datasets.

Dataset	Weeplaces				CAMRa2011				Douban			
Metric	N@20	N@50	R@20	R@50	N@20	N@50	R@20	R@50	N@20	N@50	R@20	R@50
Baseline recommender												
Popular	0.063	0.074	0.126	0.176	0.099	0.122	0.172	0.226	0.003	0.005	0.009	0.018
NeuMF	0.193	0.244	0.271	0.295	0.305	0.367	0.393	0.450	0.124	0.167	0.248	0.316
Attention-based group recommender												
AGREE	0.224	0.267	0.354	0.671	0.307	0.418	0.529	0.688	0.201	0.224	0.297	0.488
MoSAN	0.287	0.334	0.548	0.738	0.423	0.466	0.572	0.801	0.163	0.209	0.384	0.459
SIGR	0.357	0.391	0.524	0.756	0.499	0.524	0.585	0.825	0.217	0.235	0.436	0.560
GroupIM	0.431	0.456	0.575	0.773	0.637	0.659	0.753	0.874	0.257	0.284	0.523	0.696
HHGR	0.379	0.398	0.554	0.764	0.517	0.532	0.703	0.830	0.254	0.267	0.507	0.677
S^2-HHGR	**0.456**	**0.478**	**0.592**	**0.797**	**0.645**	**0.671**	**0.779**	**0.883**	**0.279**	**0.294**	**0.561**	**0.741**

- **AGREE**: It utilizes **attentional preference aggregation** to compute group member and learns the group-item interaction.
- **MoSAN**: It employs a collection of **sub-attentional networks** to learn each user's preference and models member interactions.
- **SIGR**: It introduces a latent variable and **the attention mechanism** to learn users'local and global social influence. It also utilizes **the bipartite graph** embedding model to alleviate the data sparsity problem.
- **GroupIM**: It **maximizes the mutual information** between the user representations and its belonged group representations to alleviate the data sparsity problem.

Experiments: Performance on Sparsity Datasets

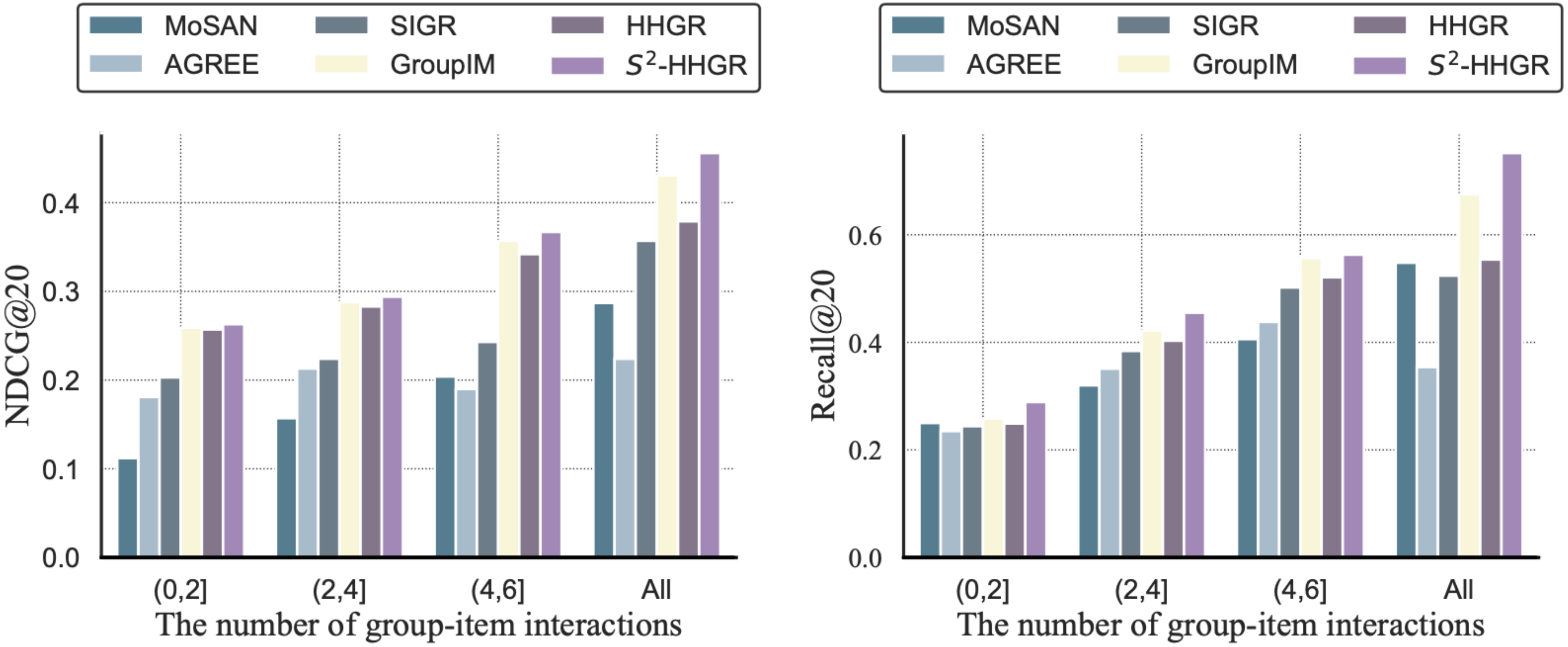


Figure 3: Performance comparison of attention-based group recommendation models on sparsity datasets.

Experiments: Ablation Study

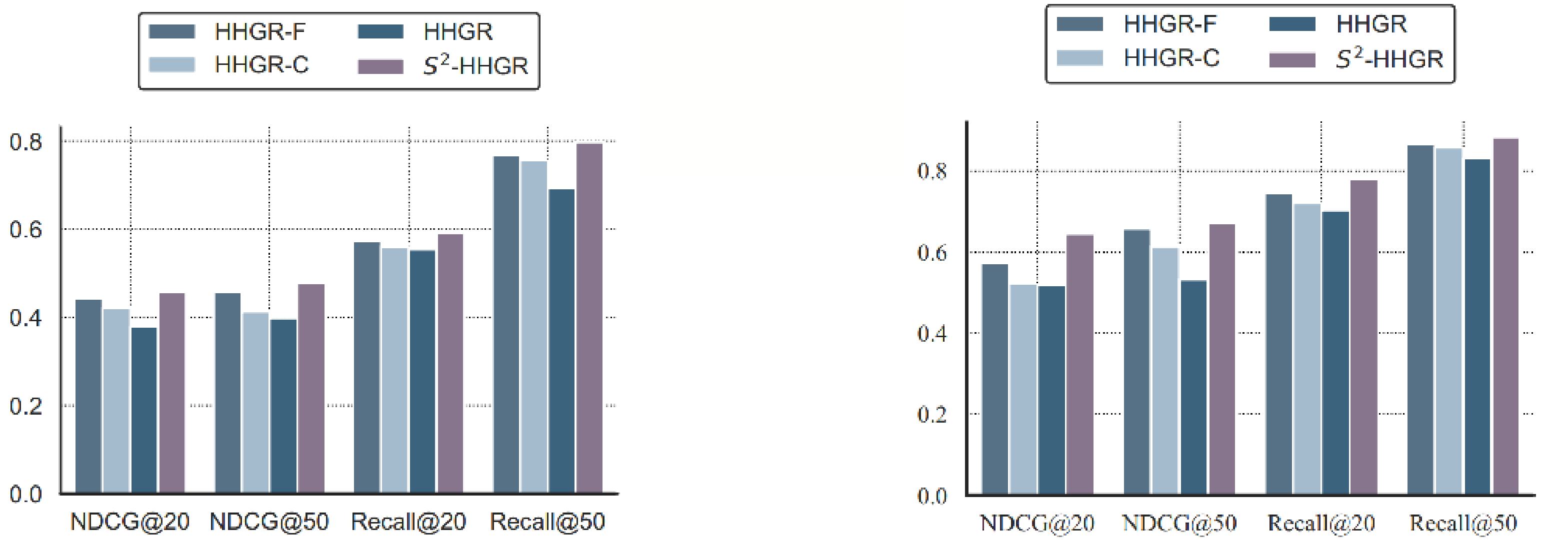
- **Investigation of the hierarchical hypergraph**

Table 3: Comparison between HHGR and its variants.

Method	Weeplaces		CAMRa2011	
Metric	N@50	R@50	N@50	R@50
HHGR-wu	0.288	0.683	0.495	0.797
HHGR-wg	0.378	0.751	0.511	0.815
HHGR	0.398	0.764	0.532	0.830

- **Investigation of the self-supervised learning**

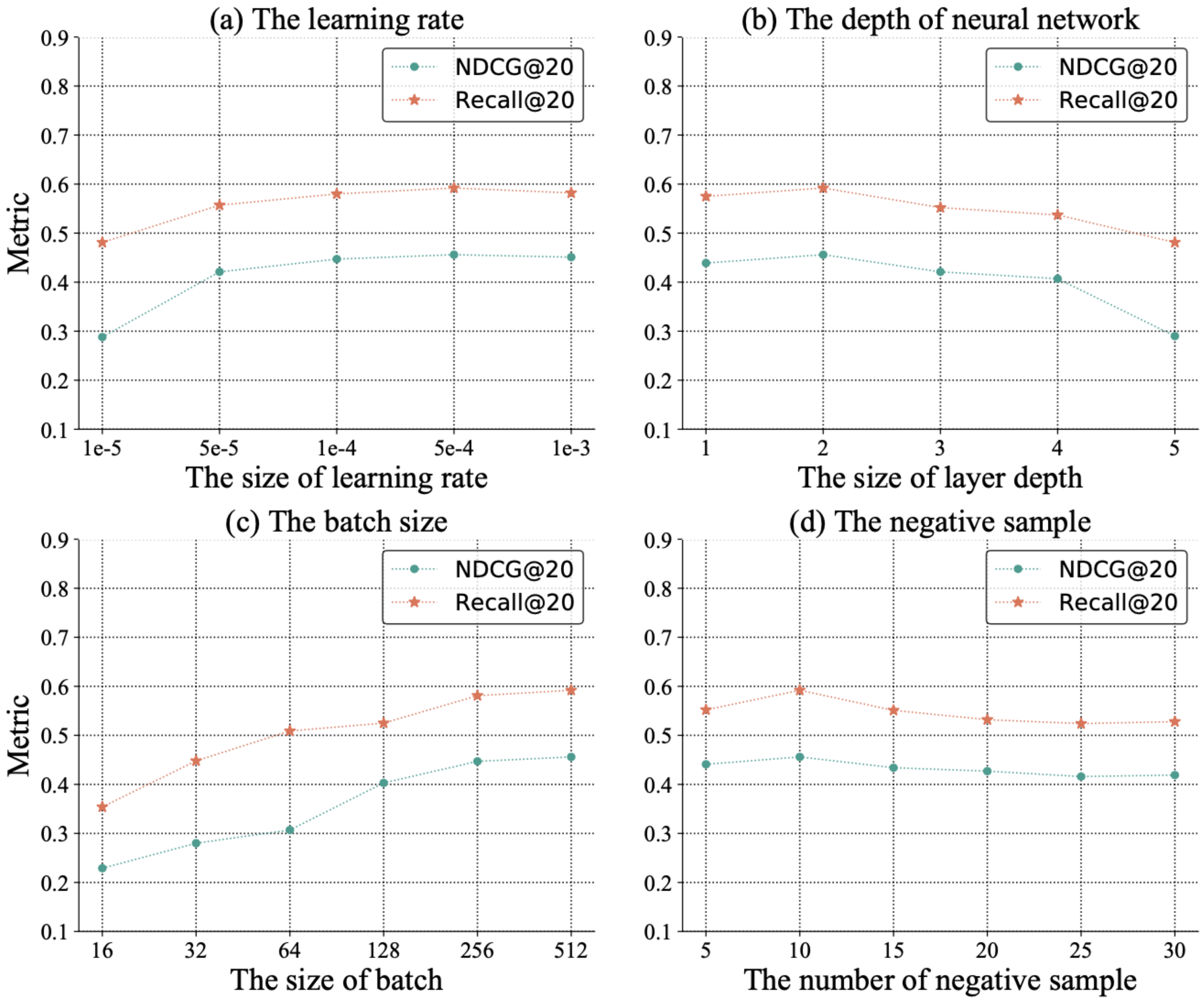
Figure 4: The influence of different self-supervised learning strategies.



(a) Weeplaces dataset

(b) CAMRa2011 dataset

Experiments: Parameter Sensitivity Analysis



Contribution

- We devise a **hierarchical hypergraph learning framework** to **capture the intra- and inter-group interactions among users**.

- We propose a **SSL strategy with different granularities** to enhance user and group representations and **alleviate the data sparsity problem**.

- We **conduct extensive experiments on three group recommendation datasets** to exhibit the superiority of the proposed model.

Conclusion and Future Work

Conclusion

- ❒ We devise a **hierarchical hypergraph learning framework**.
- ❒ We propose **a SSL strategy with different granularities**.
- ❒ We conduct **extensive experiments on three group recommendation datasets**.

Future Work

- ❒ **deepen the application of self-supervised learning in group recommendation models**
- ❒ **design more general auxiliary tasks for the recommendation**

Thanks for your listening！

jw.zhang@cqu.edu.cn