LREC-COLING 2024

# Hypergraph-Based Session Modeling: A Multi-Collaborative Self-Supervised Approach for Enhanced Recommender Systems


**Xiangping Zheng[1,3,5], Bo Wu[2*], Alex X. Zhang[4], Wei Li[1,3,5]**

1. College of Computer Science and Technology, Harbin Engineering University
2. R&D and External Relations Department, Xiangjiang Laboratory
3. Modeling and Emulation in E-Government National Engineering Laboratory
4. Harvest Fund Management Co.,Ltd.
5. Qingdao Innovation and Development Base of Harbin Engineering University

# CONTENTS

# 1 Background

# 1. Background



- In recent times, consumers have increasingly favored online product selection over in-person retail experiences. However, this shift brings with it the challenge of information overload.
- Session-based recommendation methods, renowned for their high practical relevance, are specifically engineered to delineate user intent by an by analyzing the behavioral sequences of users within sessions.
- GNN-based approaches, map each session to a subgraph respectively, and then use the subgraph as the input of GNN to further capture the dependencies of nodes in the subgraph to provide suggestions for the next project.

# 1. Background



Although these GNN-based models have exhibited promising performance in SBR, two pivotal issues warrant further in-depth investigation:

- **1) Modeling High-Order Item Relations**: In GNN-based approaches, item data is commonly interconnected via pairwise relationships. Nonetheless, real-world transactions frequently feature complex item structures characterized by high-order interconnections. Consequently, the development of a more generalized architecture that proficiently learns representations of items in higher-order relationships becomes vital.

- **2) Mitigating Item Data Sparsity:** Data sparsity in item interactions is a widespread problem in practical recommendation scenarios, primarily due to limited user interactions with a large number of items. Therefore, it is crucial to explore innovative SSL approaches to improve item representations and address data sparsity in session-based recommendation.

# 2 Method

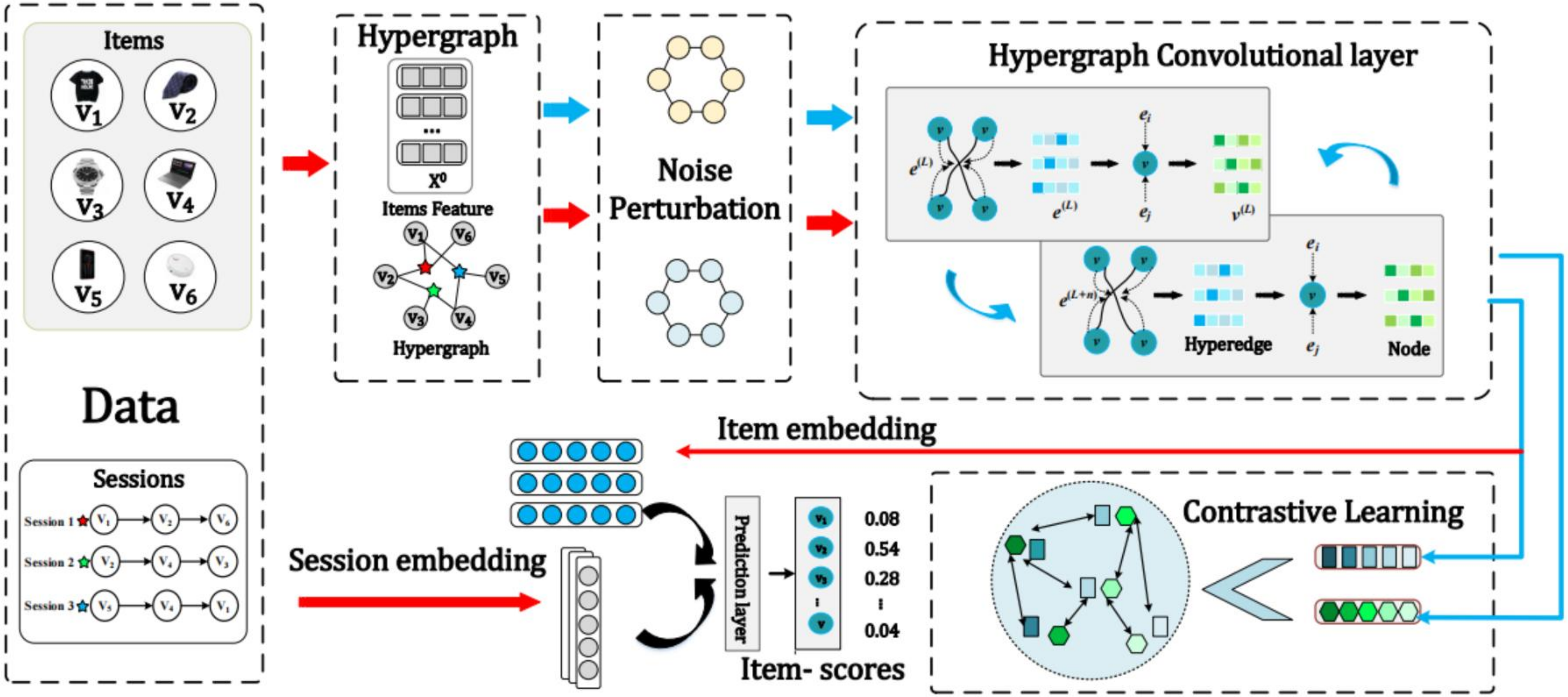
# 2. Method



Motivated by the aforementioned discussions, **we present a novel approach named Multi collaborative self-supervised learning in hypergraph neural networks, or Mssen, designed to explore user intent**.

- **1)** To tackle the first challenge, which revolves around capturing high-order relations among sessions, we introduce an innovative hypergraph modeling technique.
- **2)** Addressing the second challenge related to data sparsity, we delve into the realm of self-supervised learning (SSL) within the context of item-session hypergraphs.
- **3)** By jointly optimizing these tasks, we observe substantial improvements in recommendation performance.

Our Mssen consists of two critical tasks: one is the main task for the recommendation, and the other is SSL acted as the auxiliary task to boost the former.



## 2.1 Hypergraph Network for SBR Task

**Hypergraph Construction.** In our exploration of Session-based Recommendation systems, we embrace an advanced hypergraph structure. We map sessions to expansive hyperedges. Each hyperedge in this structure has the capability to interlink an extensive array of vertices. Moreover, while graphs typically encounter difficulties in representing the multifaceted semantic relationships of items across diverse sessions, our hypergraph paradigm adeptly captures these variable semantic linkages.

**Hypergraph Convolutional Network.** After hypergraph construction, we further develop a hypergraph neural network (HGNN) to capture the item-level high-order relations. We concatenate the hyperedge groups to generate the hypergraph incidence matrix.

$$\mathbf{X}^{(l+1)} = \mathbf{Q}\mathbf{X}^{(l)}\Theta^{(l)}, \mathbf{Q} = \hat{\mathbf{D}}^{-1}\mathbf{H}\mathbf{W}\mathbf{B}^{-1}\mathbf{H}^{T}$$

Here, hypergraph convolution can be conceptualized as a two-stage process, involving a "nodes-hyperedges nodes" feature transformation, which effectively refines features based on the hypergraph structure. Specifically, item features are initially aggregated according to the hyperedges, resulting in hyperedge features obtained by multiplying the transpose of the matrix (**Stage 1: nodes to hyperedges**). Subsequently, the final node features are derived by aggregating their respective related hyperedge features, achieved through the multiplication of the matrix (**Stage 2: hyperedges to nodes**).

### 2.1 Hypergraph Network for SBR Task

**Recommendation Generation.** Embracing insights derived from the methodology in SR-GNN, we enhance the embedding process for a given session s. Acknowledging the variable significance of the embedded information, we integrate a soft-attention mechanism designed to more accurately encapsulate the representational quality of items within a session.

We employ a cross-entropy-based loss function for every session graph to quantify the disparity between our model's predictions and the actual sequence outcomes.

$$\mathcal{L}_t = -\sum_{i=1}^{N} \mathbf{y}_i log(\hat{\mathbf{y}}_i) + (1 - \mathbf{y}_i) log(1 - \hat{\mathbf{y}}_i)$$

where y is the one-hot encoding vector of the ground truth.

## 2.2 Enhancing SBR with Self-Supervised Task

In comparison to other recommendation paradigms, session-based recommendation is particularly vulnerable to the challenge of data sparsity, primarily due to the limited short-term interactions. Additionally, while hypergraph modeling has demonstrated substantial improvements in performance, we hypothesize that the inherent issue of data sparsity might impede the full potential of hypergraph modeling, ultimately leading to suboptimal recommendation results. Drawing inspiration from the successful applications of self-supervised learning in graph-related tasks, which have shown promise in addressing data sparsity concerns, we introduce an innovative integration of self-supervised learning into the network to enhance the performance of session-based recommendation.

### 2.2 Enhancing SBR with Self-Supervised Task

**Noise Perturbation.** We propose two types of noise augmentation strategies: multiplicative noise and additive noise. These strategies involve introducing noise directly into the representation, which proves to be effective for generating diverse views of the data.

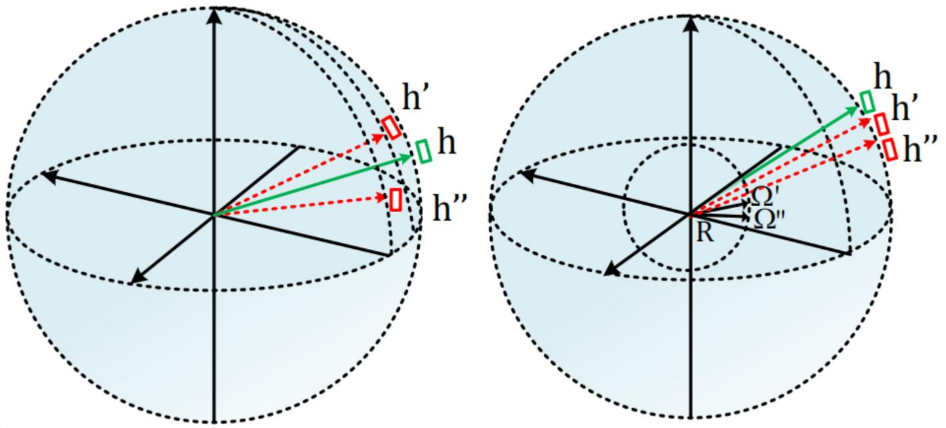
**Multiplicative Noise.**

$$\mathbf{H}' = \mathbf{H} \odot \mathbf{N}, \mathbf{H}'' = \mathbf{H} \odot \mathbf{N}$$

**Additive Noise.**

$$\mathbf{H}' = \mathbf{H} + \boldsymbol{\Omega}' \odot \mathbf{sign}(\mathbf{H}), \mathbf{H}'' = \mathbf{H} + \boldsymbol{\Omega}'' \odot \mathbf{sign}(\mathbf{H})$$

Embedding Space




(a) Multiplicative Noise (b) Additive Noise

### 2.2 Enhancing SBR with Self-Supervised Task

**Creating another self-supervised signal.** In the pursuit of reinforcing the supervised learning signals, we use a dual hypergraph Infomax (DHI) following the mechanism Hyperedge-to-Node (H2N), and drawing inspiration from the principles outlined in deep graph infomax (DGI) .

To elaborate, within the designated hypergraph layer, denoted as L, we denote the features of hyperedges. It bears mention that hypergraph convolution unfolds across two pivotal phases involving a nodes–hyperedges–nodes transformation of features. Initially, we procure the hyperedge-specific features, followed by garnering those pertaining to hypernodes. The inherent mechanics of hypergraph encoding serve to amalgamate insights from neighbors exhibiting structural semblance, thereby offering an efficient methodology to pinpoint neighboring sessions or items.

### 2.3 Model Optimization

Finally, we unify the recommendation task and this self-supervised task into a primary&auxiliary learning framework, where the former is the primary task and the latter is the auxiliary task. Formally, the joint learning objective:

$$\mathcal{L}_{LOSS} = \mathcal{L}_t + \alpha \mathcal{L}_{SSL}$$

where α is a learnable weight to control the magnitude of the self-supervised task. It should be noted that, we jointly optimize the two throughout the training.

# 3 Experiment

## 3.1 Experimental Settings

**Datasets.** For the verification of our methodology, we employed five authentic benchmark datasets (please refer to Table 1 for more details). These datasets include Tmall, Nowplaying, Diginetica, all of which are frequently exploited for testing session-based recommendation approaches.

| Statistics | Tmall | Nowplaying | Diginetica |
|---|---|---|---|
| # Sessions (Training) | 351,268 | 825,304 | 719,470 |
| # Sessions (Testing) | 25,898 | 89,824 | 60,858 |
| # Items | 40,728 | 60,417 | 43,097 |
| Avg. Length of Sessions | 6.69 | 7.42 | 5.12 |

Table 1. Dataset and graph statistics.

**Evaluation Metrics.** As recommender systems can only recommend a few items at once, the actual item a user might pick should be amongst the first few items of the list. We adopt two widely used ranking based metrics: P@K and MRR@K by following previous works.Specifically, we mainly choose to use top-10 and top-20 items to evaluate a recommender system.

**Parameter Settings**. We standardized the embedding dimension at 100, alongside consistently setting the batch size at 100 across all experimental models. We applied L2 regularization at a rate of $10^{-5}$ and controlled for the hyper-parameters among the models to ensure equitable benchmarking. Within our approach, the weighting matrices originate from a normal distribution N.

### 3.2 Overall Comparison

| Methods | | Tmall | | | | Nowplaying | | | | Diginetica | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | P@10 | MRR@10 | P@20 | MRR@20 | P@10 | MRR@10 | P@20 | MRR@20 | P@10 | MRR@10 | P@20 | MRR@20 |
| Traditional | Item-KNN | 6.65 | 3.11 | 9.15 | 3.31 | 10.96 | 4.55 | 15.94 | 4.91 | 25.07 | 10.77 | 35.75 | 11.57 |
| | FPMC | 13.10 | 7.12 | 16.06 | 7.32 | 5.28 | 2.68 | 7.36 | 2.82 | 15.43 | 6.20 | 26.53 | 6.95 |
| RNNs | GRU4Rec | 9.47 | 5.78 | 10.93 | 5.89 | 6.74 | 4.40 | 7.92 | 4.48 | 17.93 | 7.33 | 29.45 | 8.33 |
| | NARM | 19.17 | 10.42 | 23.30 | 10.70 | 13.60 | 6.62 | 18.59 | 6.93 | 35.44 | 15.13 | 49.70 | 16.17 |
| | STAMP | 22.63 | 13.12 | 26.47 | 13.36 | 13.22 | 6.57 | 17.66 | 6.88 | 33.98 | 14.26 | 45.64 | 14.32 |
| GNNs | FGNN | 20.67 | 10.07 | 25.24 | 10.39 | 13.89 | 6.80 | 18.78 | 7.15 | 37.72 | 15.95 | 50.58 | 16.84 |
| | GCE-GNN | 28.01 | 15.08 | 33.42 | 15.42 | 16.94 | 8.03 | 22.37 | 8.40 | 41.16 | 18.15 | 54.22 | 19.04 |
| | $S^2$-DHCN | 26.22 | 14.60 | 31.42 | 15.05 | 17.35 | 7.87 | 23.50 | 8.18 | 40.21 | 17.59 | 53.66 | 18.51 |
| | **Ours** | **33.53** | **18.98** | **38.51** | **19.60** | **18.22** | **9.35** | **24.11** | **9.81** | **42.33** | **19.88** | **55.17** | **19.64** |

- **Consistently Strong Performance:** Across all datasets and metrics, our proposed model consistently outperformed the existing baselines. This robust performance underscores the effectiveness of our approach. Notably, even on the Tmall dataset, where existing baselines had already achieved high performance, our method managed to push the performance boundary further.

- **Remarkable Performance Compared to Traditional and RNN Methods:** Our model exhibited remarkable performance when compared to traditional and RNN-based approaches. This suggests that our approach, which converts sequential item transitions into graph-structured data to capture the inherent order of item-transition patterns, delivers superior results.

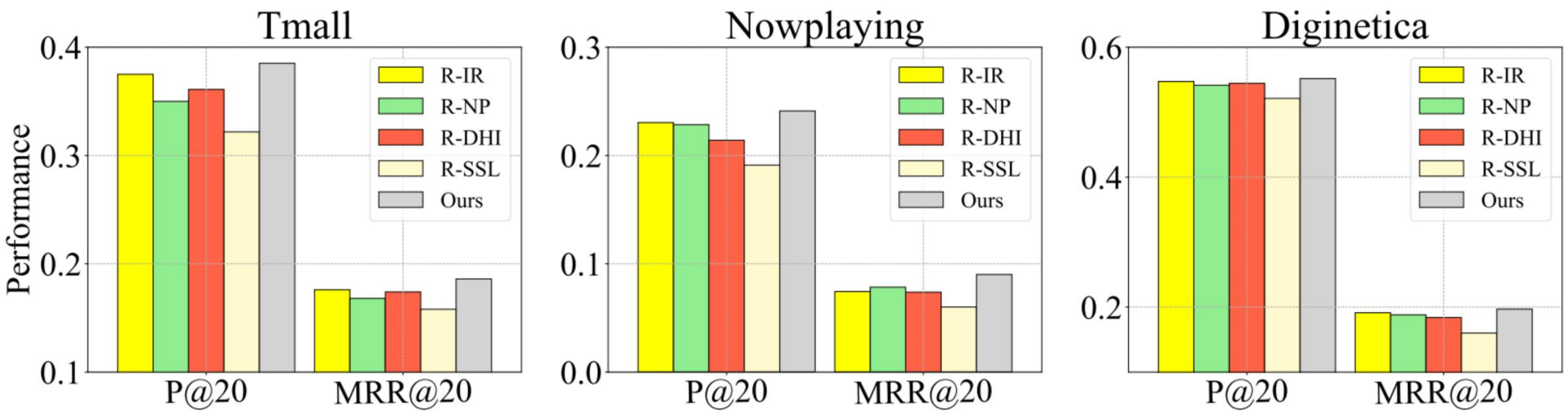
## 3.2 Overall Comparison

| Methods | | Tmall | | | | Nowplaying | | | | Diginetica | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | P@10 | MRR@10 | P@20 | MRR@20 | P@10 | MRR@10 | P@20 | MRR@20 | P@10 | MRR@10 | P@20 | MRR@20 |
| Traditional | Item-KNN | 6.65 | 3.11 | 9.15 | 3.31 | 10.96 | 4.55 | 15.94 | 4.91 | 25.07 | 10.77 | 35.75 | 11.57 |
| | FPMC | 13.10 | 7.12 | 16.06 | 7.32 | 5.28 | 2.68 | 7.36 | 2.82 | 15.43 | 6.20 | 26.53 | 6.95 |
| RNNs | GRU4Rec | 9.47 | 5.78 | 10.93 | 5.89 | 6.74 | 4.40 | 7.92 | 4.48 | 17.93 | 7.33 | 29.45 | 8.33 |
| | NARM | 19.17 | 10.42 | 23.30 | 10.70 | 13.60 | 6.62 | 18.59 | 6.93 | 35.44 | 15.13 | 49.70 | 16.17 |
| | STAMP | 22.63 | 13.12 | 26.47 | 13.36 | 13.22 | 6.57 | 17.66 | 6.88 | 33.98 | 14.26 | 45.64 | 14.32 |
| GNNs | FGNN | 20.67 | 10.07 | 25.24 | 10.39 | 13.89 | 6.80 | 18.78 | 7.15 | 37.72 | 15.95 | 50.58 | 16.84 |
| | GCE-GNN | 28.01 | 15.08 | 33.42 | 15.42 | 16.94 | 8.03 | 22.37 | 8.40 | 41.16 | 18.15 | 54.22 | 19.04 |
| | $S^2$-DHCN | 26.22 | 14.60 | 31.42 | 15.05 | 17.35 | 7.87 | 23.50 | 8.18 | 40.21 | 17.59 | 53.66 | 18.51 |
| | **Ours** | **33.53** | **18.98** | **38.51** | **19.60** | **18.22** | **9.35** | **24.11** | **9.81** | **42.33** | **19.88** | **55.17** | **19.64** |

• **Competitive Results Against Graph-Based Baselines:** Our method also achieved competitive results when compared to graph-based baselines. The improvements in our method primarily stem from our innovative contrastive learning strategies. By constructing hyperedge-level and node-level contrastive objectives to focus on fine-grained supervised signals, we enhance the learning process. This is in contrast to which employs two types of hypergraphs for sessions embedding in contrastive learning, potentially resulting in weaker signals.

### 3.3 Ablation Study

We conduct experiments to investigate the contribution of each component in our model. Specially, we design four variant versions: R-IR: We remove the initial residual for each hypergraph layer when high-order relations propagation. R-NP: We remove the noise perturbation (NP) for constructing self-supervised signals. R-DHI: We remove the dual hypergraph Infomax (DHI) for constructing self-supervised signals. R-SSL: We remove all SSL signals.




Each component contributes significantly to the model's overall performance, with self-supervised contrastive learning playing a central role in driving the improvements. The use of two contrastive objectives, the initial residual technique, and the overall model design contribute to the superior performance of our approach in session-based recommendation.

# 4 Conclusion

# 4. Coclusion



This paper introduces a novel approach, Mssen, for session-based recommendation, addressing data sparsity by employing SSL on item-session hypergraphs. Extensive empirical evaluations consistently show its superiority over existing methods. It’s worth noting that hypergraph modeling in session-based recommendation is an emerging field with broader applications in graph-related research, offering ample room for further exploration and development.

LREC-COLING 2024

哈爾濱工程大學
HARBIN ENGINEERING UNIVERSITY

湘江實驗室
XIANGJIANG LABORATORY

# Thanks