ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10097031

INT-GNN: A USER INTENTION AWARE GRAPH NEURAL NETWORK FOR SESSION-BASED RECOMMENDATION

Guangning Xu[1] *Jinyang Yang*[1] *Jinjin Guo*[2] *Zhichao Huang*[2*] *Bowen Zhang*[3*]

[1]Computer Science & Technology, Harbin Institute of Technology, Shenzhen, China
[2]JD Intelligent Cities Research, JD iCity, Beijing, China
[3]College of Big Data and Internet, Shenzhen Technology University, Shenzhen, China

ABSTRACT

Session-Based Recommendation (SBR) is a spotlight research problem. Although many efforts have been made, challenges still exist. The key to unlocking this shackle is the user intention, an intuitive but hard-to-model concept in the anonymous session. Unlike previous research, we suggest mining potential user intention by counting the number of item occurrences in a user session and considering the long interval between item re-interactions. Beyond these, we take user preference, a biased user intention, into account in the prediction stage. Forming these together, we propose a model named user **Int**ention aware **G**raph **N**eural **N**etwork (Int-GNN) aiming at capturing user intention. Extensive experiments have been conducted on three real-world datasets, and the results show the superiority of our method. The code is available on GitHub: `https://github.com/xuguangning1218/IntGNN_ICASSP2023`

***Index Terms*—** session based recommendation, user intention, number of item occurrences, GNN

1. INTRODUCTION

Recently, a new pearl has attracted people's eyeballs: the Session-Based Recommendation (SBR) [1, 2]. It focuses on performing recommendations under anonymous, ad hoc session conditions, which shows more challenge than other recommendation tasks. It can be roughly divided into machine learning methods and deep learning methods.

The former ones take charge in the early stages of research by mining item-to-item relationships [3, 4] overlooked user interaction behaviors. The latter ones show up like bamboo shoots after rain. They mainly consist of three parts. First, most RNN-based methods model user intention through chronological item transitions [5, 6], which may not be entirely applicable for the SBR [7]. Second, the attention-based methods [8, 9] attempt to discriminate the user's interest in items suffering from missing the partial sequence information. Third, the GNN-based methods have shed some light on the SBR task. They capture items transitions relationships within a session to obtain a robust item latent representation [10, 11], yet the over-smoothing issue might lead to a limited user intention capture. Furthermore, the global graph-based methods [12, 13] are proposed for performance improvement while they are resources costing.

Although the above works are fascinating, our work attempts to expand the horizon of the SBR task from a new perspective. Researches [14, 15] show that user intention is helpful. Due to privacy policies, we have limited ability to model user intention from user interaction, e.g., clicking, browsing, and purchasing. To fill this gap, we propose a novel method named a user **Int**ention Aware **G**raph **N**eural **N**etwork (Int-GNN) aiming at capturing user intention in three aspects. By introducing the number of item occurrences, an item shows up times in a session, we design the Item Occurrence Graph Neural Network (IO-GNN) for the user interaction intention. To model a user's long-interval re-interactions, we proposed the Interactive Position Graph Neural Network (IP-GNN). Following a famous psychological claim, the anchoring effect [16], people rely heavily on biased information while making decisions, we proposed Multi-Scores Generator to combine both item characteristics and user preference. The main contributions of this work can be summarized as follows: i) To the best of our knowledge, it is the first time to introduce the number of item occurrences for mining user intention without accessing private data. ii) We proposed the Int-GNN, which considers the number of item occurrences, item re-interaction interval, and user preference in the SBR. iii) Extensive experiments have been conducted on three commonly used real-world datasets.

2. PRELIMINARY

Problem Formulation. Let a set $V = \{v_1, v_2, \ldots, v_{|V|}\}$ denote items involved in all sessions and a vector $\mathbf{S} = [v_1, v_2, \ldots, v_n]$ denotes an anonymous session with n interactions, where $v_i \in V$ means the i-th interaction item. The target of the SBR is to estimate a probability $\hat{y}_i$ of the potential interaction item, where $\hat{y}_i$ is the i-th value of a probability

*To whom correspondence should be addressed (iceshzc@gmail.com; zhang_bo_wen@foxmail.com).

vector $\hat{\mathbf{y}} = \left[\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_{|V|}\right]$. Formally, the objective function is presented: $\hat{y}_i(\theta^*) = \max_\theta P\{v_{n+1}|v_1, v_2, \ldots, v_n, \theta\}$, where θ denotes learnable parameters. **Item Occurrence and Sequential Item Occurrence**. Let $\mathbf{S} = \left[v_s^1, v_s^2, \ldots, v_s^n\right]$ be a input session vector and $\mathbf{S}_i = \left[v_s^1, v_s^2, \ldots, v_s^i\right]$ be the sub-vector of $\mathbf{S}$, where $i \leq n$. We use an aggregate function called $\text{Cnt}(v_s^i, \mathbf{S}_i)$ to count the number of occurrence for item v_s^i in $\mathbf{S}_i$. To measure the occurrence order of $\mathbf{S}$, the sequential item occurrence vector is defined: $\text{seqOcc}(\mathbf{S}) = \left[\text{Cnt}(v_s^1, \mathbf{S}_1), \ldots, \text{Cnt}(v_s^n, \mathbf{S}_n)\right]$.

3. METHOD

3.1. Model Overview

We first give a bird's-eye view of the Int-GNN architecture shown in Fig. 1. The Int-GNN is mainly composed of three components, namely, the IO-GNN, the IP-GNN, and the Multi-Scores Generator. First, a specific session is fed into the IO-GNN and IP-GNN to obtain the hidden user intention. After that, the Multi-Scores Generator looks forward to two inputs. One comes from the number of item occurrences aimed at identifying user preference from a session. The other is the fusion information from the IO-GNN and IP-GNN, which generates the items' characteristics representation. Finally, two derived scores contribute their endeavors to the final prediction.

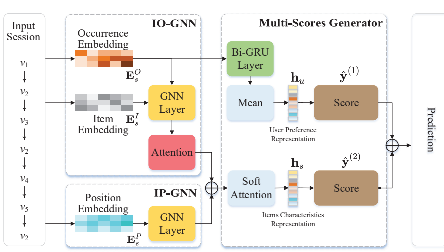

Fig. 1. Overview of the proposed Int-GNN

3.2. Item Occurrence Graph Neural Network

Global Item Occurrence Embedding As the number of item occurrences in a session exhibits user intention to some extent, we design the global item occurrence embedding to capture this characteristic. To begin with, the maximum number of item occurrences is anchored to M. Then, we introduce the global item occurrence embedding: $\mathbf{E}^O = [\mathbf{e}_0^o, \mathbf{e}_1^o, \mathbf{e}_2^o, \ldots, \mathbf{e}_M^o]^\top$, where an embedding vector $\mathbf{e}_i^o \in \mathbb{R}^d$ is a learnable vector, and it can be utilized by all items that show up in i times in a session, d is the dimension of the embedding. After that, we define a lookup operation to identify the specific embedding vector. For an item v_i occurs $\text{Cnt}(v_s^i, \mathbf{S})$ times in the session $\mathbf{S}$, the corresponding embedding vector is set to $\mathbf{e}^o_{\text{Cnt}(v_s^i, \mathbf{S})}$. Finally, we can get the session-based item occurrence embedding $\mathbf{E}_s^O = \left[\mathbf{e}^o_{\text{Cnt}(v_s^1,\mathbf{S})}, \mathbf{e}^o_{\text{Cnt}(v_s^2,\mathbf{S})}, \ldots, \mathbf{e}^o_{\text{Cnt}(v_s^{|V_s|},\mathbf{S})}\right]^\top$ for a specific session $\mathbf{S}$. Obviously, many occurrences of an item explicate that the corresponding item is a potential recommendation outcome. In this way, the global item occurrence embedding $\mathbf{E}^O$ can model the user intention for the given session by providing the item occurrence information.

Prorogation on GNN To cooperate with the number of item occurrences, we further employ a GNN to model user intention hiding inside item transitions. First, we establish a session directed graph $G_s = (V_s, E_s)$ according to [10], where $V_s \subseteq V$ is the unique-item set in a session. Node $v_i \in V_s$ denotes an item and an edge $(v_{i-1}, v_i) \in E_s$ represents the transition from item v_i to item v_{i-1}. To describe the session directed graph G_s, we introduce two adjacent matrices $\mathbf{A}_s^{\text{in}}, \mathbf{A}_s^{\text{out}} \in \mathbb{R}^{|V_s|\times|V_s|}$. Note that $\mathbf{A}_s^{\text{in}}$ and $\mathbf{A}_s^{\text{out}}$ are self-loop and degree normalized. Second, we generate item embedding $\mathbf{E}^I$ according to [5, 10]. After a lookup operation, we get the session-based item embedding $\mathbf{E}_s^I$. Third, we conduct message passing : $\mathbf{Z}_s^{\text{in}} = \mathbf{A}_s^{\text{in}}\left(\mathbf{E}_s^I\mathbf{W}_1 + \mathbf{E}_s^O\right), \mathbf{Z}_s^{\text{out}} = \mathbf{A}_s^{\text{out}}\left(\mathbf{E}_s^I\mathbf{W}_2 + \mathbf{E}_s^O\right), \mathbf{Z}_s' = \mathbf{Z}_s^{\text{in}} + \mathbf{Z}_s^{\text{out}} + \mathbf{B}$, where $\mathbf{W}_1, \mathbf{W}_2, \mathbf{B} \in \mathbb{R}^{d\times d}$ are learnable parameters, $\mathbf{Z}_s' \in \mathbb{R}^{|V_s|\times d}$ is gathered into $\mathbb{R}^{n\times d}$ [10].

Sequential Item Occurrence Attention As the user intention are chronological and undulating, we propose the Sequential Item Occurrence Attention for modelling user intention in time dimension. First, we utilize the $\text{seqOcc}(\mathbf{S})$ to construct the sequential item occurrence embedding in a session. Similar to the generation of the $\mathbf{E}_s^O$, we have the session-based sequential item occurrence embedding: $\mathbf{E}_s^{SI} = \left[\mathbf{e}^{si}_{\text{Cnt}(v_s^1,\mathbf{S}_1)}, \mathbf{e}^{si}_{\text{Cnt}(v_s^2,\mathbf{S}_2)}, \ldots, \mathbf{e}^{si}_{\text{Cnt}(v_s^n,\mathbf{S}_n)}\right]^\top$. Secondly, for a chronological session sub-vector $\mathbf{S}_i$, an attention value is calculated: $\alpha_i = \frac{\sigma(\mathbf{Z}'_{s,i}\mathbf{W}+\mathbf{b})\cdot\mathbf{e}^{si}_{\text{Cnt}(v_s^i,\mathbf{S}_i)}}{\|\mathbf{e}^{si}_{\text{Cnt}(v_s^i,\mathbf{S}_i)}\|_2}$, where $\mathbf{Z}'_{s,i}$ is the i-th row of the output $\mathbf{Z}_s'$, $\mathbf{W} \in \mathbb{R}^{d\times d}$ are learnable weights, $\mathbf{b} \in \mathbb{R}^d$ are learnable bias, $\sigma(\cdot)$ is the Sigmoid function. Last, we assign the above attention values $\boldsymbol{\alpha}$ to the GNN output $\mathbf{Z}_s'$ and generate the final output of Sequential Item Occurrence Attention: $\mathbf{Z}_{s,i} = \alpha_i \cdot \mathbf{Z}'_{s,i}$.

3.3. Interactive Position Graph Neural Network

As aforementioned, a re-interaction after a long interval of steps might reflect a user intention. It is better to tackle this problem by position embedding. However, existing position embedding methods treat each position independently, which limits the capacity to handle the long interval steps re-interaction issues. In the end, we propose to equip the position embedding with a GNN to fix this issue. For a start, the maximum length for a session is set to N. Then, we introduce the position embedding $\mathbf{E}^P = [\mathbf{e}_1^p, \mathbf{e}_2^p, \ldots, \mathbf{e}_N^p]^\top$. Its lookup result is denoted as $\mathbf{E}_s^P = [\mathbf{e}_1^p, \mathbf{e}_2^p, \ldots, \mathbf{e}_n^p]^\top$. After that, we propose to construct an session-based items' co-occurrence graph $G_p = (P, E_p)$, where $P = \{1, 2, \dots, N\}$ denotes the reversed position index set, E_p denotes an edge set. The corresponding adjacent matrix $\mathbf{A}$ is given: $\mathbf{A}_{i,j} = 1, v_i = v_j$ otherwise $\mathbf{A}_{i,j} = 0$, where $i, j \in P$. In this case, a re-occurred item interaction will lead to the neighbor position nodes connecting. Finally, the propagation of the IP-GNN is formulated: $\mathbf{Z}_p = \mathbf{A}\mathbf{E}_s^P\mathbf{W} + \mathbf{B}$, where $\mathbf{W}, \mathbf{B} \in \mathbb{R}^{d\times d}$ are learnable parameters.

3.4. Multi-Scores Generator

To enhance the prediction, we proposed a multi-score output considering both user preference and item characteristics.

As for the first score, we suppose that the distribution of the session-based number of item occurrences can indicate the ephemeral user preference. In this case, we propose to generate a user preference representation $\mathbf{h}_u \in \mathbb{R}^d$ from sequential item occurrence embedding $\mathbf{E}_s^{SI} \in \mathbb{R}^{n\times d}$ by using series of operations composed of Bi-GRU, FC, and Mean. Then, the user preference embedding is given: $\mathbf{E}_u^O = \left[\mathbf{e}^o_{\text{Cnt}(v_s^1,\mathbf{S})}, \mathbf{e}^o_{\text{Cnt}(v_s^2,\mathbf{S})}, \dots, \mathbf{e}^o_{\text{Cnt}(v_s^{|V|},\mathbf{S})}\right]^\top$, where $\mathbf{E}_u^O \in \mathbb{R}^{|V|\times d}$ denotes the user preference over the whole item set V. Finally, we generate the first output: $\hat{\mathbf{y}}^{(1)} = \widetilde{\mathbf{E}}_u^O\left(\widetilde{\mathbf{h}}_u\right)^\top$, where $\widetilde{\mathbf{h}}_u$ denote the L_2-norm of $\mathbf{h}_u$, and $\widetilde{\mathbf{E}}_u^O$ is the L_2-norm of $\mathbf{E}_u^O$. Note that the user preference is upon the distribution of number of item occurrences instead of item characteristics. As for the second score, we measure how item characteristics contribute to the prediction. We first present the soft-attention to generate the item characteristics representation $\mathbf{h}_s = \sum_{i=1}^{n} \frac{\mathbf{q}^\top\sigma((\mathbf{Z}_{s,i}+\mathbf{Z}_{p,i})\mathbf{W}+\mathbf{b})}{\|\mathbf{q}\|}\mathbf{E}_{s,i}^I$, where $\mathbf{E}_s^I \in \mathbb{R}^{n\times d}$ are session-based item embedding, $\mathbf{W} \in \mathbb{R}^{d\times d}$, $\mathbf{b}, \mathbf{q} \in \mathbb{R}^d$ are learnable parameters, $\sigma(\cdot)$ is the Sigmoid function. Finally, we get output according to [17] : $\hat{\mathbf{y}}^{(2)} = \mu\widetilde{\mathbf{E}}^I\left(\widetilde{\mathbf{h}}_s\right)^\top$, where μ is a temperature hyper-parameter, $\widetilde{\mathbf{h}}_s$ is a L_2-norm of $\mathbf{h}_s$, and $\widetilde{\mathbf{E}}^I$ is a L_2-norm of $\mathbf{E}^I$. The final probability output is generated by considering both user preference and item characteristics: $\hat{\mathbf{y}} = \text{SoftMax}\left(\hat{\mathbf{y}}^{(1)} + \hat{\mathbf{y}}^{(2)}\right)$.

4. EXPERIMENT

4.1. Data and Experiment Settings

We chose three commonly used datasets as our benchmark: (1) **Diginetica** comes from the CIKM Cup 2016, we used its transaction data, (2) **Tmall** is released by the IJCAI-15 competition. It consists of anonymous users' shopping logs on Tmall, and (3) **RetailRocket** records users' browsing activities for six months in the real world. For the preprocessing steps, we followed the procedures as [13, 18, 19], e.g., we dropped the sessions whose length is less than 2 (3 for the RetailRocket), and we also abandoned items that occur less than 5. Following previous methods [19, 13], we set the dimension of latent vectors to $d = 100$, the mini-batch was 512, and cross-entropy loss. All model parameters were initialized by $\mathcal{N}(0, 0.1)$ Gaussian distribution. The learning rate for Adam [20] was set to 0.00128, and its decay ratio was set to 0.3 after every three epochs. Moreover, we set the maximum occurrence value to $M = 300$ and the maximum length of any session to $N = 100$. The temperature hyper-parameter $\mu = 12.5$. All the experiments of Int-GNN were conducted in 5 runs and reported their mean and standard deviation.

4.2. Baselines and Metrics

To evaluate the performance, we comprehensively compared IntGNN with typical methods and state-of-the-art models, a total of 11 baselines: (1) **POP** recommends the top-N frequent items based on the training set; (2) **GRU4Rec** [5] utilizes GRU for sequential information; (3) **NARM** [2] employs RNNs and attention mechanisms to generate session embedding; (4) **STAMP** [8] generates session embedding with the long- and short-term memory; (5) **SR-GNN** [10] utilizes GNNs to capture items transitions; (6) **NISER** [17] introduces L_2 norm to solve the long-tail issue; (7) **LESSR** [21] uses GRU on graph structure to learn item-level embedding; (8) **GCE-GNN** [12] introduces a global graph and combines it with the session graph for learning item embedding; (9) **DSAN** [18] applies an adaptively sparse transformation function and target attention to eliminate the unrelated items; (10) **DHCN** [13] introduces a hypergraph convolutional network that integrates a self-supervised method for session embedding; (11) **COTREC** [19] co-trains two distinct graph encoders in a self-supervised manner, aiming at building the internal and external connectivity of the session. Two commonly used evaluation metrics, named P@N and MRR@N, were adopted in our experiments. The P@N evaluates the proportion of correct recommendations in an unranked list, and the MRR@N considers the position of correct recommended items in a ranked list.

4.3. Overall Comparison

To evaluate the overall performance, Table 1 summarizes results over three real-world datasets. From this table, we can draw the following conclusions. First, the deep learning based methods outperform the traditional methods. Second, the GNN-based methods outperform others since they can model complex item transitions. Third, our Int-GNN model shows a superior performance gain among the GNN-based methods in most cases. It might be that these methods concentrate on short-term or long-term item transitions paying less attention to the overall characteristics of a session. Unlike the above methods, the Int-GNN takes advantage of user intention by using a newly introduced number of item occurrences, which is more intuitive for the recommendation.

Table 1. Overall performance comparison over three real-world datasets

Datasets	Diginetica				Tmall				RetailRocket			
Metrics	P@20	MRR@20	P@10	MRR@10	P@20	MRR@20	P@10	MRR@10	P@20	MRR@20	P@10	MRR@10
POP	1.18	0.28	0.76	0.26	2.00	0.90	1.67	0.88	1.12	0.30	0.61	0.27
GRU4Rec [5](2016)	29.45	8.33	17.93	7.33	10.93	5.89	9.47	5.78	44.01	23.67	38.35	23.27
NARM [2](2017)	49.70	16.17	35.44	15.13	23.30	10.70	19.17	10.42	50.22	24.59	42.07	24.88
STAMP [8](2018)	45.64	14.32	33.98	14.26	26.47	13.36	22.63	13.12	50.96	25.17	42.95	24.61
SR-GNN [10](2019)	50.73	17.59	36.86	15.52	27.57	13.72	23.41	13.45	50.32	26.57	43.21	26.07
NISER[17](2019)	53.39	18.72	40.20	17.82	33.79	16.67	28.46	16.38	54.90	29.89	47.69	29.38
LESSR [21](2020)	51.71	18.15	36.16	15.64	27.88	12.08	22.68	11.68	53.05	28.01	45.76	27.51
GCE-GNN [12](2020)	54.22	19.04	41.16	18.15	33.42	15.42	28.01	15.08	50.60	25.39	43.53	24.89
DSAN [18](2021)	53.76	18.99	40.29	18.05	36.45	18.17	30.91	**17.76**	56.54	30.74	49.05	30.21
DHCN [13](2021)	53.18	18.44	39.87	17.53	31.42	15.05	26.22	14.60	53.66	27.30	46.15	26.85
COTREC [19](2021)	54.18	19.07	**41.88**	18.16	36.35	18.04	30.62	17.65	56.17	29.97	48.61	29.46
Int-GNN	**55.16**±0.04	**19.46**±0.02	41.84±0.06	**18.53**±0.02	**40.77**±0.16	**18.20**±0.12	**34.28**±0.12	17.74±0.12	**58.02**±0.08	**31.48**±0.11	**50.41**±0.08	**30.94**±0.11
Improvement	1.7%	1.9%	-0.1%	2.0%	11.86%	0.27%	10.91%	-0.09%	2.61%	2.39%	2.77%	2.42%

Table 2. Ablation study on three datasets

Datasets	Diginetica				Tmall				RetailRocket			
Metrics	P@20	MRR@20	P@10	MRR@10	P@20	MRR@20	P@10	MRR@10	P@20	MRR@20	P@10	MRR@10
Int-GNN	55.16±0.04	**19.46**±0.02	**41.84**±0.06	**18.53**±0.02	40.77±0.16	18.20±0.12	34.28±0.12	17.74±0.12	**58.02**±0.08	31.48±0.11	**50.41**±0.08	30.94±0.11
w/o-IO-GNN	54.94±0.05	19.32±0.03	41.60±0.03	18.39±0.03	40.33±0.12	19.08±0.09	34.07±0.18	18.64±0.09	57.88±0.09	**31.57**±0.10	50.30±0.10	**31.05**±0.10
w/o-IP-GNN	52.23±0.08	18.07±0.03	39.07±0.14	17.15±0.04	**40.87**±0.13	18.36±0.17	**34.43**±0.27	17.91±0.18	54.04±0.17	28.67±0.10	46.79±0.15	28.16±0.09
w/o-MScore	**55.19**±0.07	19.26±0.02	41.78±0.05	18.33±0.02	39.54±0.22	18.24±0.11	32.98±0.15	17.78±0.11	57.68±0.09	31.08±0.02	49.93±0.15	30.54±0.03
w/o-IO-IP	52.00±0.05	18.27±0.02	39.00±0.05	17.37±0.02	40.28±0.11	19.01±0.05	34.18±0.09	18.59±0.06	53.51±0.09	28.78±0.08	46.46±0.08	28.29±0.07
w/o-IO-MScore	54.92±0.05	19.17±0.02	41.61±0.06	18.25±0.02	39.38±0.04	18.91±0.06	33.22±0.11	18.48±0.06	57.27±0.10	30.86±0.10	49.56±0.10	30.33±0.10
w/o-IP-MScore	52.34±0.07	18.03±0.03	39.02±0.05	17.10±0.04	39.63±0.11	18.31±0.10	33.10±0.11	17.86±0.10	53.52±0.11	28.01±0.06	46.11±0.11	27.49±0.07
vanilla-GNN	52.11±0.06	18.21±0.01	38.95±0.05	17.29±0.01	39.18±0.22	**19.36**±0.05	33.14±0.11	**18.94**±0.05	53.19±0.13	28.33±0.07	45.87±0.05	27.81±0.07

4.4. Ablation study

Here, we conduct a complete ablation study on the proposed modules, where the vanilla-GNN method is unequipped for all proposed modules. Note that we retained item occurrence embedding in w/o-IO-GNN as an input for the Multi-Scores Generator, and w/o-MScores was obtained by removing user preference embedding from the Int-GNN. Table 2 shows the results. Three significant observations can be made. First, performance declines in most cases when we ablate any proposed module on Diginetica and RetailRocket. As for the Tmall, it resists the negative effect because it is generated from a biased sampling strategy. Second, the pair-ablation results illustrate a manifest performance decrease over three datasets. In particular, we get an extinct result when we abandon both IP-GNN and IO-GNN, supporting the effectiveness of the newly introduced item occurrence concept. Third, vanilla-GNN shows the worse results, denoting that the proposed modules certainly earn a performance gain.

4.5. The analysis of IP-GNN

As we first proposed to use GNN in position embedding, we compared the IP-GNN to non-GNN methods, such as the vanilla position embedding (vanilla-Pos) [9, 17, 22], and the reversed position embedding (reversed-Pos) [12, 13, 19]. The results, shown in Fig. 2, indicate that the Int-GNN equipped with IP-GNN shows a mind-blowing performance compared with vanilla-Pos and reversed-Pos on the Diginetica and RetailRocket datasets. The reason might be that vanilla-Pos and reversed-Pos treat interactive positions independently, while the IP-GNN aggregates the corresponding potential positions. As for the Tmall, it is insensitive to IP-GNN, which its biased sampling strategy might cause. In this manner, the IP-GNN provides valuable information on how long a user takes to re-interact with the same item.

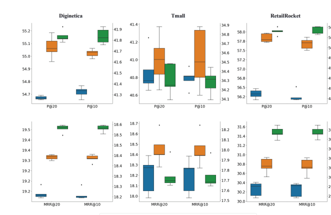

Fig. 2. Comparison of different position embedding methods

5. CONCLUSION

In this paper, we proposed utilizing the number of item occurrences to give insight into user intention capturing. Upon this concept, we model user intention by the Int-GNN model, which includes the IO-GNN, the IP-GNN, and the Multi-Scores Generator by considering user intention in the number of item occurrences, user intention in item re-interaction intervals, and user preference, respectively. Extensive experiments show the superiority of the Int-GNN.

6. ACKNOWLEDGEMENT

This research is supported by the Stable Support Project for Shenzhen Higher Education Institutions (SZWD2021011) and the Research Promotion Project of Key Construction Discipline in Guangdong Province (2022ZDJS112).

7. REFERENCES

[1] Dietmar Jannach, Malte Ludewig, and Lukas Lerche, "Session-based item recommendation in e-commerce: on short-term intents, reminders, trends and discounts," *User Model. User Adapt. Interact.*, vol. 27, no. 3-5, pp. 351–392, 2017.

[2] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma, "Neural attentive session-based recommendation," in *CIKM*, 2017, pp. 1419–1428.

[3] Badrul Munir Sarwar, George Karypis, Joseph A. Konstan, and John Riedl, "Item-based collaborative filtering recommendation algorithms," in *WWW*, 2001, pp. 285–295.

[4] Guy Shani, Ronen I. Brafman, and David Heckerman, "An mdp-based recommender system," in *UAI*, 2002, pp. 453–460.

[5] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk, "Session-based recommendations with recurrent neural networks," in *ICLR*, 2016.

[6] Huizhao Wang, Guanfeng Liu, An Liu, Zhixu Li, and Kai Zheng, "DMRAN: A hierarchical fine-grained attention-based network for recommendation," in *IJCAI*, 2019, pp. 3698–3704.

[7] Shoujin Wang, Longbing Cao, Yan Wang, Quan Z. Sheng, Mehmet A. Orgun, and Defu Lian, "A survey on session-based recommender systems," *ACM Comput. Surv.*, vol. 54, no. 7, pp. 154:1–154:38, 2022.

[8] Qiao Liu, Yifu Zeng, Refuoe Mokhosi, and Haibin Zhang, "STAMP: short-term attention/memory priority model for session-based recommendation," in *SIGKDD*, Yike Guo and Faisal Farooq, Eds., 2018, pp. 1831–1839.

[9] Wang-Cheng Kang and Julian J. McAuley, "Self-attentive sequential recommendation," in *ICDM*, 2018, pp. 197–206.

[10] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan, "Session-based recommendation with graph neural networks," in *AAAI*, 2019, pp. 346–353.

[11] Chengfeng Xu, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Fuzhen Zhuang, and et al., "Graph contextualized self-attention network for session-based recommendation," in *IJCAI*, 2019, pp. 3940–3946.

[12] Ziyang Wang, Wei Wei, Gao Cong, Xiaoli Li, Xianling Mao, and Minghui Qiu, "Global context enhanced graph neural networks for session-based recommendation," in *SIGIR*, 2020, pp. 169–178.

[13] Xin Xia, Hongzhi Yin, Junliang Yu, Qinyong Wang, Lizhen Cui, and Xiangliang Zhang, "Self-supervised hypergraph convolutional networks for session-based recommendation," in *AAAI*, 2021, pp. 4503–4511.

[14] Nengjun Zhu, Jian Cao, Yanchi Liu, Yang Yang, Haochao Ying, and Hui Xiong, "Sequential modeling of hierarchical user intention and preference for next-item recommendation," in *WSDM*, 2020, pp. 807–815.

[15] Nuo Xu, Pinghui Wang, Long Chen, Jing Tao, and Junzhou Zhao, "MR-GNN: multi-resolution and dual graph neural network for predicting structured entity interactions," in *IJCAI*, 2019, pp. 3968–3974.

[16] Adrian Furnham and Hua Chu Boo, "A literature review of the anchoring effect," *The Jour. of Socio-eco.*, vol. 40, no. 1, pp. 35–42, 2011.

[17] Priyanka Gupta, Diksha Garg, Pankaj Malhotra, Lovekesh Vig, and Gautam Shroff, "Niser: Normalized item and session representations with graph neural networks," in *CIKM Workshop*, 2019.

[18] Jiahao Yuan, Zihan Song, Mingyou Sun, Xiaoling Wang, and Wayne Xin Zhao, "Dual sparse attention network for session-based recommendation," in *AAAI*, 2021, pp. 4635–4643.

[19] Xin Xia, Hongzhi Yin, Junliang Yu, Yingxia Shao, and Lizhen Cui, "Self-supervised graph co-training for session-based recommendation," in *CIKM*, 2021, pp. 2180–2190.

[20] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[21] Tianwen Chen and Raymond Chi-Wing Wong, "Handling information loss of graph neural networks for session-based recommendation," in *KDD*, 2020, pp. 1172–1180.

[22] Feng Wang, Xiang Xiang, Jian Cheng, and Alan Loddon Yuille, "Normface: L_2 hypersphere embedding for face verification," in *MM*, 2017, pp. 1041–1049.