A Domain-aware Attentive Graph Convolution Network for Shared-account Cross-domain Sequential Recommendation

DA-GCN

Ho In Jung

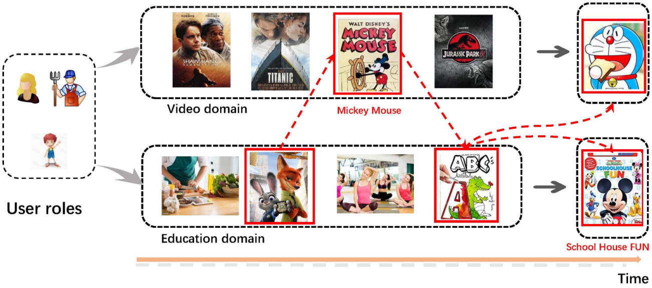
Department of Computational Science & Technology
Seoul National University

July 19, 2021

Overview

Concept

- Cross-domain Sequential Recommendation for Shared-account



- With H latent users $(U_{k,1}, U_{k,2}, \cdots, U_{k,h}, \cdots, U_{k,H})$ under each account (U_k)

Preliminaries

- Behavior sequence from domain A and B $S_A = \{A_1, A_2, \cdots, A_i\}$ and
  $S_B = \{B_1, B_2, \cdots, B_i\}$
  where $A_i \in \mathcal{A}(1 \leq i \leq p)$ is the item index in domain A; $\mathcal{A}$ is the item set in domain A
  where $A_i \in \mathcal{B}(1 \leq i \leq q)$ is the item index in domain B; $\mathcal{B}$ is the item set in domain B

  $U = \{U_1, U_2, \cdots, U_k, \cdots, U_{|U|}\}$ is a subset of accounts,
  where $U_k \in \mathcal{U}(1 \leq k \leq n)$ denotes the account index and $\mathcal{U}$, representing the whole account set.

- The recommendation probabilities for all candidate items in domain A and B are :

$$P(A_{i+1}|S_A, S_B) \sim f_A(S_A, S_B); \tag{1}$$

$$P(B_{j+1}|S_B, S_A) \sim f_B(S_B, S_A), \tag{2}$$

  where $P(A_{i+1}|S_A, S_B)$ is the probability of recommending the next item A_{i+1} in domain A.
  $f_A(S_A, S_B)$ is the learned function to estimate $P(A_{i+1}|S_A, S_B)$.

Representation Learning with Latent Users

- behaviors under the same account are usually generated by different users and different interests.
- The number and identity of all users are both **unknown**
  H latent users $(U_{k,1}, U_{k,2}, \cdots, U_{k,h}, \cdots, U_{k,H})$ under each account (U_k),
  and the embedding of user $(U_{k,h})$ is denoted by $\mathbf{e}_{U_{k,h}} \in \mathbb{R}^d$
  $\Rightarrow$ learned by accumulating all the information from connected items in domain A and B.

Notation Table

Variable	**Definition**
A_i, B_j	item neighbors of $U_{k,h}$ in domain A and B
$\mathcal{N}_{A_i}^{U_{k,h}} \mathcal{N}_{B_j}^{U_{k,h}}$	item neighbor set of $U_{k,h}$ in domain A and B
$m_{U_{k,h} \leftarrow A_i}, m_{U_{k,h} \leftarrow B_j}$	message representation
$\gamma_{A_i}^{U_{k,h}}$	learnable parameter between A_i and $U_{k,h}$
$\gamma_{B_j}^{U_{k,h}}$	learnable parameter between B_j and $U_{k,h}$
$\gamma_{U_{k,h}}^{U_{k,h}}$	attentive weights
$\odot$	element-wise product
$\mathbf{e}_{A_i} \in \mathbb{R}^d$	embedding vectors of item A_i
$\mathbf{e}_{b_j} \in \mathbb{R}^d$	embedding vectors of item B_i
$\mathbf{e}_{U_{k,h}}$	embedding vectors of user $U_{k,h}$
$\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{d` \times d}$	trainable weight matrices

Table: Notation table for representation learning with latent users

Message Passing

- message passed from $A_i \in \mathcal{N}_{A_i}^{U_{k,h}}$ to $U_{k,h}$

$$m_{U_{k,h} \leftarrow A_i} = \gamma_{A_i}^{U_{k,h}} (\mathbf{W}_1 \mathbf{e}_{A_i} + \mathbf{W}_2 (\mathbf{e}_{A_i} \odot \mathbf{e}_{U_{k,h}})), \tag{3}$$

- message passed from $B_j \in \mathcal{N}_{B_j}^{U_{k,h}}$ to $U_{k,h}$

$$m_{U_{k,h} \leftarrow B_j} = \gamma_{B_j}^{U_{k,h}} (\mathbf{W}_1 \mathbf{e}_{B_j} + \mathbf{W}_2 (\mathbf{e}_{B_j} \odot \mathbf{e}_{U_{k,h}})), \tag{4}$$

- self-connection to retain the information carried by the target user

$$m_{U_{k,h} \leftarrow U_{k,h}} = \gamma_{U_{k,h}}^{U_{k,h}} (\mathbf{W}_1 \mathbf{e}_{U_{k,h}}), \tag{5}$$

Domain-aware Attention Mechanism

- fully capture each latent user's distinct preferences over different interacted items
- compute the importance of different item nodes to the target user

$$s_{A_i}^{U_{k,h}} = f(\mathbf{e}_{U_{k,h}}, \mathbf{e}_{A_i}), \tag{6}$$

$$s_{B_j}^{U_{k,h}} = f(\mathbf{e}_{U_{k,h}}, \mathbf{e}_{B_j}). \tag{7}$$

where $f(\cdot)$ is a pairwise similarity metric.

- normalization operation for attentive weight on each item from domain A

$$\gamma_{A_i}^{U_{k,h}} = \exp(s_{A_i}^{U_{k,h}}) / \left(\sum_{A_i' \in \mathcal{N}_{A_i}^{U_{k,h}}} \exp(s_{A_i'}^{U_{k,h}}) + \sum_{B_j' \in \mathcal{N}_{B_j}^{U_{k,h}}} \exp(s_{B_j'}^{U_{k,h}}) + s_{U_{k,h}}^{U_{k,h}} \right) \tag{8}$$

where $s_{U_{k,h}}^{U_{k,h}}$ weights the self-connection within user $U_{k,h}$

Message Aggregation

- The embedding of user $U_{k,h}$, $\mathbf{e}_{U_{k,h}}$ is then updated via :

$$\mathbf{e}_{U_{k,h}} = \text{LeakyReLU}\left(\mathbf{m}_{U_{k,h} \leftarrow U_{k,h}} + \sum_{A_i \in \mathcal{N}_{A_i}^{U_{k,h}}} \mathbf{m}_{U_{k,h} \leftarrow A_i} + \sum_{B_j \in \mathcal{N}_{B_j}^{U_{k,h}}} \mathbf{m}_{U_{k,h} \leftarrow B_j}\right) \tag{9}$$

  where all the message passed from users and items in both domains are aggregated.

- account-level representation U_k for all latent user's representations

$$\mathbf{e}_{U_k} = \frac{1}{H} \sum_{h=1}^{H} \mathbf{e}_{U_{k,h}}. \tag{10}$$

User-specific Item Representation Learning

- Item representation is learned by aggregating the information from two types of nodes.
- i.e., the connected users and items within the same domain
- we take domain A as an example $\rightarrow$ applicable for domain B
- varied relevance of linked users/itemes to the target item
- assess the importance of the connected users and the items that have **sequential relationships** to the target item

Notation Table

Variable	Definition
$A_i, U_{k,h}$	connected item and user nodes
$\mathcal{N}_{A_{i-1}}^{A_i}$	item neighbor set of A_i in domain A
$\mathcal{N}_{U_{k,h}}^{A_i}$	user neighbor set of A_i in domain A
$m_{A_i \leftarrow A_{i-1}}$	message passed from A_{i-1} to A_i
$m_{A_i \leftarrow U_{k,h}}$	message passed from user $U_{k,h}$ to A_i
$m_{A_i \leftarrow A_i}$	retained message of A_i
$\gamma_{A_{i-1}}^{A_i}$	control parameter how much information can be passed
$\gamma_{U_{k,h}}^{A_i}, \gamma_{A_i}^{A_i}$	learnable parameter

Table: Notation table for user-specific representation learning

Message Passing

- message passed from $A_{i-1} \in \mathcal{N}_{A_{i-1}}^{A_i}$ to A_i

$$m_{A_i \leftarrow A_{i-1}} = \gamma_{A_{i-1}}^{A_i}(\mathbf{W}_1 \mathbf{e}_{A_{i-1}} + \mathbf{W}_2(\mathbf{e}_{A_{i-1}} \odot \mathbf{e}_{A_i})), \tag{11}$$

- message passed from user $U_{k,h} \in \mathcal{N}_{A_i}^{U_{k,h}}$ to A_i

$$m_{A_i \leftarrow U_{k,h}} = \gamma_{U_{k,h}}^{A_i}(\mathbf{W}_1 \mathbf{e}_{U_{k,h}} + \mathbf{W}_2(\mathbf{e}_{U_{k,h}} \odot \mathbf{e}_{A_i})), \tag{12}$$

- retained message of A_i

$$m_{A_i \leftarrow A_i} = \gamma_{A_i}^{A_i}(\mathbf{W}_1 \mathbf{e}_{A_i}), \tag{13}$$

Sequential-aware Attention Mechanism

- The importance of item A_{i-1} to A_i

$$s_{A_{i-1}}^{A_i} = f(\mathbf{e}_{A_{i-1}}, \mathbf{e}_{A_i}), \tag{14}$$

- The importance of user $U_{k,h}$ on item A_i

$$s_{A_i}^{U_{k,h}} = f(\mathbf{e}_{A_i}, \mathbf{e}_{U_{k,h}}). \tag{15}$$

- Normalization of $s_{A_{i-1}}^{A_i}$

$$\gamma_{A_{i-1}}^{A_i} = \exp(s_{A_{i-1}}^{A_i}) / \left(\sum_{A'_{i-1} \in \mathcal{N}_{A_{i-1}}^{A_i}} \exp(s_{A_{i-1}'}^{A_i}) + \sum_{U'_{k,h} \in \mathcal{N}_{U_{k,h}}^{A_i}} \exp(s_{U_{k,h}'}^{A_i}) + s_{A_i}^{A_i} \right) \tag{16}$$

where $s_{A_i}^{A_i}$ is the importance of the self-connection within item A_i.

- $\gamma_{U_{k,h}}^{A_i}, \gamma_{A_i}^{A_i}$ are obtained in the same way

Message Aggregation

- The user-specific item representation is updated by aggregating all messages passed from the neighbor user/item nodes within domain A

$$\mathbf{e}_{A_i} = \text{LeakyReLU}\left(\mathbf{m}_{A_i \leftarrow A_i} + \sum_{A_{i-1} \in \mathcal{N}_{A_{i-1}}^{A_i}} \mathbf{m}_{A_i \leftarrow A_{i-1}} + \sum_{U_{k,h} \in \mathcal{N}_{U_{k,h}}^{A_i}} \mathbf{m}_{A_i \leftarrow U_{k,h}}\right) \tag{17}$$

- And we use $g_{A_i}^h = e_{A_i}$ to denote the embedding of item A_i w.r.t the h-th user.
- The final representation of item $A_i(g_{A_i})$

$$\mathbf{g}_{A_i} = \frac{1}{H} \sum_{h=1}^{H} \mathbf{g}_{A_i^h}. \tag{18}$$

Matrix-form Propagation Rule

- propagation rule in a layer-wise matrix-form

$$E_l = \sigma((\mathcal{L} + \mathrm{I})E_{l-1}W_1 + \mathcal{L}E_{l-1} \odot E_{l-1}W_2) \tag{19}$$

  where I denotes an identity matrix; $E_l \in \mathbb{R}^{(p+n+q)\times d}$ is the representation of all user and item nodes in domain A and B

- $\mathcal{L} \in \mathbb{R}^{(p+n+q)\times(p+n+q)}$ is the Laplacian matrix for the CDS graph

$$\mathcal{L} = \begin{bmatrix} Y_{A_iA_{i-1}} & Y_{A_iU_k} & 0 \\ Y_{U_kA_i} & 0 & Y_{U_kB_j} \\ 0 & Y_{B_jU_k} & Y_{B_jB_{j-1}} \end{bmatrix} \tag{20}$$

  $Y_{A_iA_{i-1}} \in \mathbb{R}^{p\times p}$: attention matrices weights from item neighbors to the target item in domain A.

The Prediction Layer

- Max pooling operation on the item embedding within the sequence $(g_{A_1}, g_{A_2}, \cdots g_{A_i})$ (or $(g_{B_1}, g_{B_2}, \cdots g_{B_j})$, obtaining the sequence-level embedding $h`_{S_A}$ (or $h`_{S_B}$) for S_A (or S_B)

$$P(A_{i+1}|S_A, S_B) = softmax(W_A \cdot [h`_{S_A}, e_{U_k}]^T + b_A)$$

$$P(B_{j+1}|S_B, S_A) = softmax(W_B \cdot [h`_{S_B}, e_{U_k}]^T + b_B)$$

- negative log-likelihood loss function to train DA-GCN in each domain:

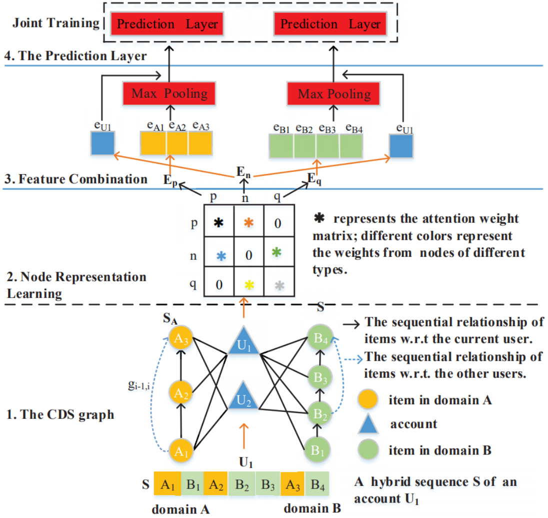
$$L_A(\theta) = -\frac{1}{|\mathbb{S}|} \sum_{S_A, S_B \in \mathbb{S}} \sum_{A_i \in S_A} \log P(A_{i+1}|S_A, S_B), \tag{21}$$

$$L_B(\theta) = -\frac{1}{|\mathbb{S}|} \sum_{S_A, S_B \in \mathbb{S}} \sum_{B_j \in S_B} \log P(B_{j+1}|S_A, S_B), \tag{22}$$

where θ represents all the parameters of DA-GCN and $\mathbb{S}$ denotes the training sequence in both domain A and B.

$$L(\theta) = L_A(\theta) + L_B(\theta) \tag{23}$$

DA-GCN


Joint Training
Prediction Layer
Prediction Layer
4. The Prediction Layer
Max Pooling
Max Pooling
e_{U1}
e_{A1} e_{A2} e_{A3}
e_{B1} e_{B2} e_{B3} e_{B4}
e_{U1}
3. Feature Combination
E_p
E_n
E_q
p n q
p
n
q
0
0
0
✱ represents the attention weight matrix; different colors represent the weights from nodes of different types.
2. Node Representation Learning
S
S_A
A_3
A_2
A_1
$g_{i-1,i}$
U_1
U_2
B_4
B_3
B_2
B_1
1. The CDS graph
U_1
The sequential relationship of items w.r.t the current user.
The sequential relationship of items w.r.t. the other users.
item in domain A
account
item in domain B
S A_1 B_1 A_2 B_2 B_3 A_3 B_4
domain A
domain B
A hybrid sequence S of an account U_1

Dataset

- HVIDEO : Education video logs& Video(TV series, movies) logs
- HAMAZON : Movie watching and rating records & Book reading and rating behaviors

TABLE I: Statistics of the datasets.

HAmazon		HVIDEO	
M-domain		*V-domain*	
#Items	67,161	#Items	16,407
#Logs	4,406,924	#Logs	227,390
B-domain		*E-domain*	
#Items	126,547	#Items	3,380
#Logs	4,287,240	#Logs	177,758
#Overlapped-users	13,724	#Overlapped-users	13,714
#Sequences	289,160	#Sequences	134,349
#Training-sequences	204,477	#Training-sequences	102,182
#Validation-sequences	49,814	#Validation-sequences	18,966
#Test-sequences	34,869	#Test-sequences	13,201

Experimental Result

Methods	HVIDEO								HAMAZON							
	E-domain (%)				V-domain (%)				M-domain (%)				B-domain (%)			
	MRR		Recall		MRR		Recall		MRR		Recall		MRR		Recall	
	@5	@20	@5	@20	@5	@20	@5	@20	@5	@20	@5	@20	@5	@20	@5	@20
POP	1.71	2.24	2.21	6.58	2.66	3.27	5.01	10.49	0.36	0.49	0.73	2.02	0.14	0.22	0.42	1.22
Item-KNN	2.11	2.90	3.01	12.11	4.43	2.93	10.48	23.93	1.28	1.86	2.58	9.00	3.23	4.55	6.65	20.94
BPR-MF	1.34	1.64	2.74	5.83	1.21	1.36	1.88	3.38	2.90	3.06	3.90	5.50	0.88	0.96	1.23	2.15
VUI-KNN	2.03	3.48	6.36	24.27	3.44	2.87	16.46	34.76	-	-	-	-	-	-	-	-
NCF-MLP++	3.92	5.14	7.36	20.81	16.25	17.90	26.10	43.04	13.68	14.21	18.44	24.31	13.67	14.05	18.14	22.08
Conet	5.01	6.21	9.26	22.71	21.25	23.28	32.94	52.72	14.64	15.12	19.25	24.46	15.85	16.28	20.98	25.56
GRU4REC	12.27	13.70	16.24	32.16	78.27	78.27	80.15	83.04	82.01	82.11	83.10	84.06	81.34	81.44	82.77	83.76
HGRU4REC	14.47	16.11	19.79	37.52	80.37	80.62	81.92	84.43	83.07	83.14	84.24	84.91	82.15	82.31	83.46	84.91
π-net	14.63	16.88	20.41	45.19	80.51	80.95	83.22	87.48	83.91	83.95	84.91	85.33	84.93	84.93	85.33	85.38
PSJNet	16.63	18.46	22.12	42.20	81.97	82.32	84.32	87.75	84.01	84.05	84.88	85.28	**85.10**	**85.11**	85.32	85.38
DA-GCN	**19.24**	**21.24**	**26.65**	**47.78**	**83.13**	**83.42**	**85.46**	**88.30**	**84.69**	**84.71**	**85.13**	**85.34**	84.81	84.81	**85.32**	**85.38**

Table 2: Experimental results on HVIDEO and HAMAZON. VUI-KNN does not work on this dataset because it needs specific time in a day which is not available on HAMAZON dataset.

Ablation Study

Variants	E-domain (%)				V-domain (%)			
	MRR		Recall		MRR		Recall	
	@5	@20	@5	@20	@5	@20	@5	@20
GCN_{OSA}	19.24	20.86	26.24	43.40	82.99	83.23	85.23	87.66
GCN_{OA}	19.11	20.81	26.07	43.78	83.09	83.35	85.43	87.88
GCN_{OS}	19.21	21.19	26.37	47.04	83.12	83.42	85.45	88.27
DA-GCN	**19.24**	**21.24**	**26.65**	**47.78**	**83.13**	**83.42**	**85.46**	**88.30**

Table 3: Ablations studies on the HVIDEO dataset.

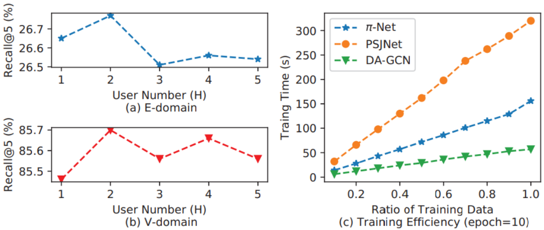


Figure 2: Impact of H and Model Training Efficiency on HVIDEO.