Conceptualize and Infer User Needs in E-commerce

ADVISOR: JIA-LING KOH
PRESENTER: CHENG-WEI CHEN
SOURCE: CIKM'19
DATE: 2021/9/27

Outline

- Introduction
  - Problem
  - Previous Work
  - Target

Problem

Historical behaviors

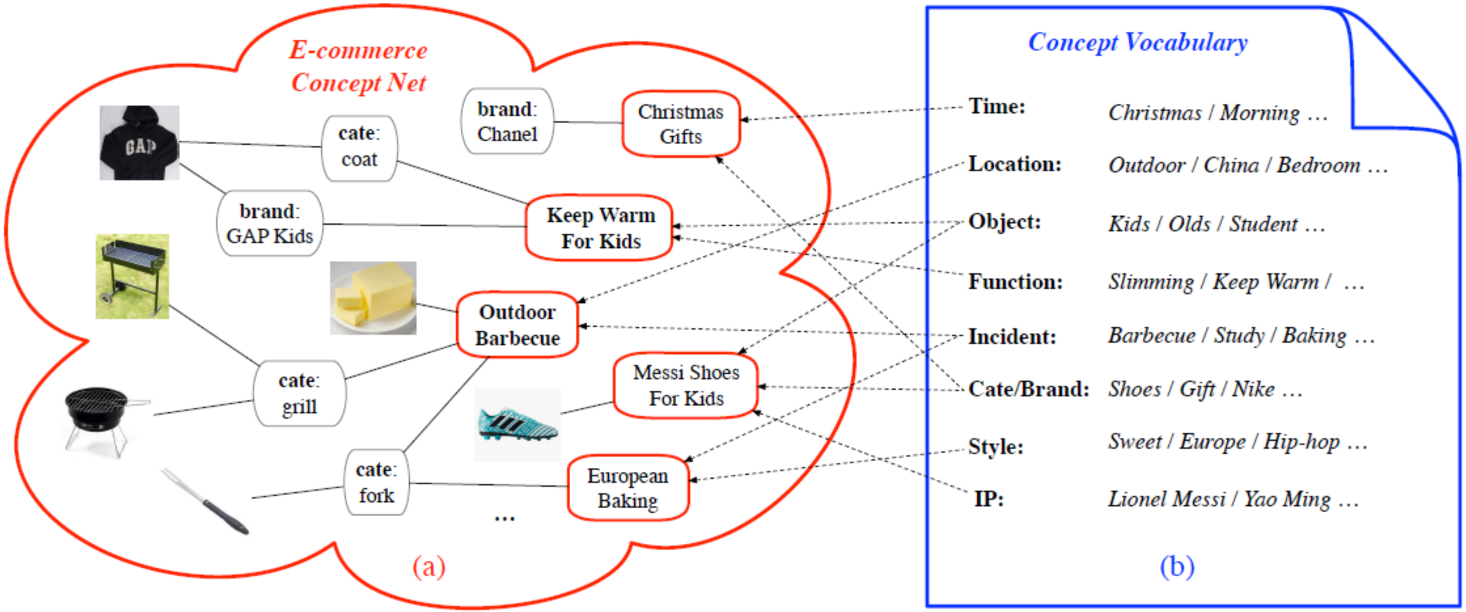
(User profile, User behaviors, Item information, etc...)

Previous work

1. Hard for the recommender system to **jump out of historical behaviors** to explore other implicit user needs
2. Current recommender systems can only **satisfy very limited user needs such as the needs for a particular category or brand**

Target

Bridging concepts connect user and items to satisfy some **high-level user needs** or shopping scenarios such as "Outdoor Barbecue" and "Keep Warm for kids"



Target

Input

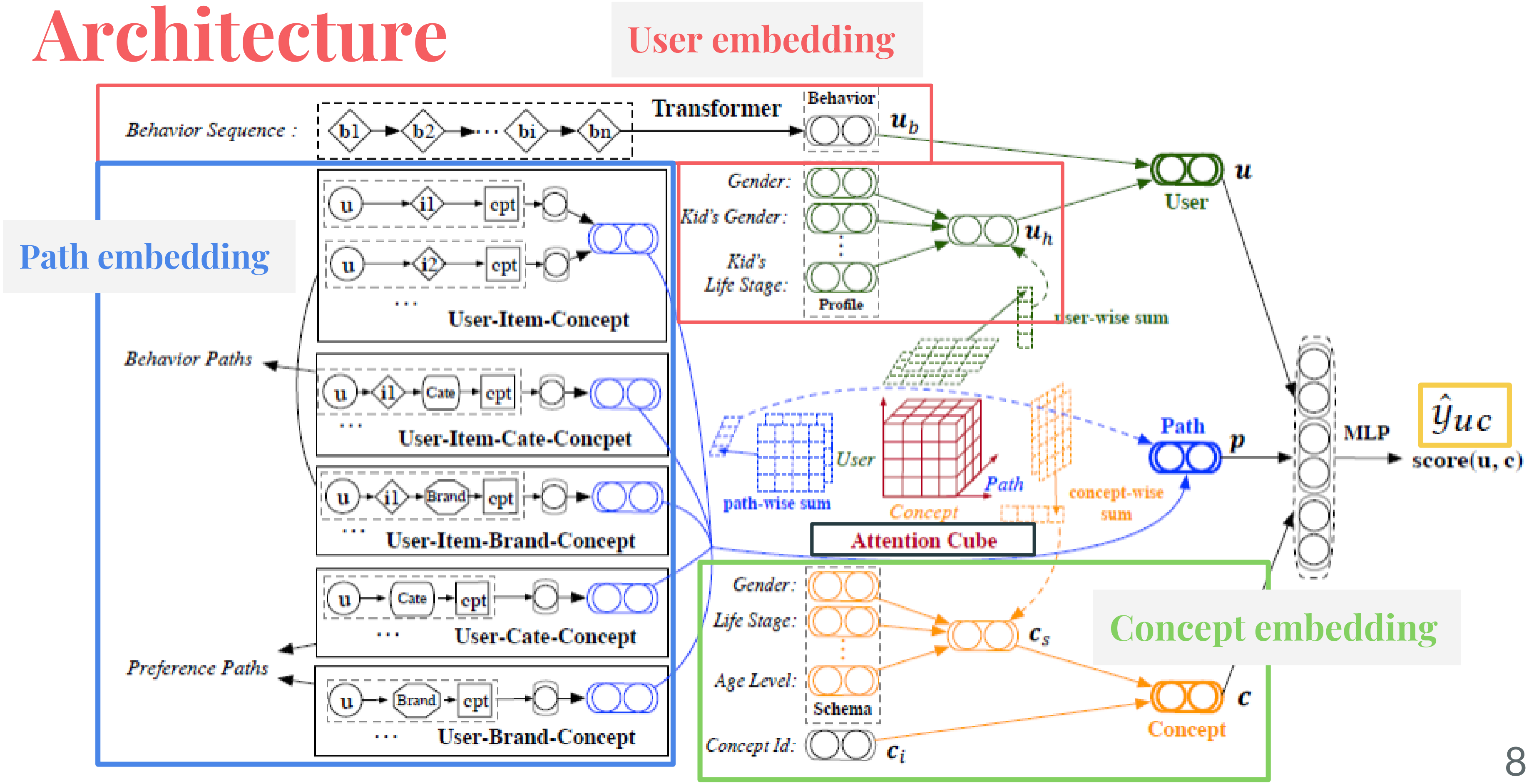
- User Historical Behavior (item, behavior type, behavior time)
- User Profile(gender, age level, kid's gender, kid's life stage, etc)
- Concept (id, gender, age, etc)
- Item(category, brand, shop,etc)
- E-commerce Concept Net

- **Output**: probability of user u will need concept c $\hat{y}_{uc}$

Outline

- Method
  - Architecture
  - Method detail
  - Loss function

Architecture



User Embedding

- User Historical Behavior $\boxed{\boldsymbol{b}_i}$ (item, behavior type, behavior time)

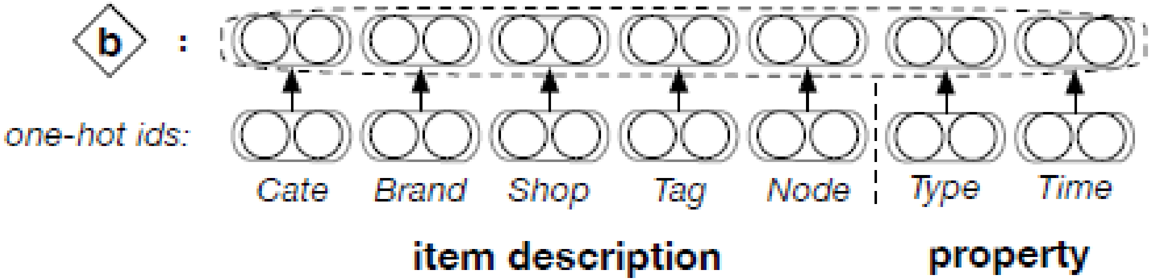


Figure 4: Encoding of user behavior

Low-dimensional of dense vector

$$\boldsymbol{b}_i = [\mathbf{W}^1_{lk}\boldsymbol{b}^1_i; \mathbf{W}^2_{lk}\boldsymbol{b}^2_i; \cdots; \mathbf{W}^F_{lk}\boldsymbol{b}^F_i],\ \mathbf{W}^f_{lk} \in \mathbf{R}^{d^f \times V^f}$$

$$\boldsymbol{u}_b = \text{Transformer}(\boldsymbol{b}_1, \boldsymbol{b}_2, \cdots, \boldsymbol{b}_n)$$

User Embedding

- User Historical Behavior $\boxed{\boldsymbol{b}_i}$ (item, behavior type, behavior time)

$$\boldsymbol{u}_b = \text{Transformer}(\boldsymbol{b}_1, \boldsymbol{b}_2, \cdots, \boldsymbol{b}_n)$$

- User Profile (gender, age level, kid's gender, kid's life stage, etc)

$$\boldsymbol{u}_h = f_u(u_h) = f_u([\boldsymbol{h}_{\text{gender}}, \boldsymbol{h}_{\text{age}}, \cdots])$$

Average pooling

$$\boldsymbol{u} = \text{FC}([\boldsymbol{u}_b; \boldsymbol{u}_h])$$

FC: Fully connected

Concept Embedding

- Concept id embedding $\boxed{\boldsymbol{c}_i}$
- Concept schema (gender, age, etc)

$$\boldsymbol{c}_s = f_c(c_s) = f_c([\boldsymbol{s}_{\text{gender}}, \boldsymbol{s}_{\text{age}}, \cdots])$$

Average pooling

$$\boldsymbol{c} = \mathbf{FC}([\boldsymbol{c}_i; \boldsymbol{c}_s])$$

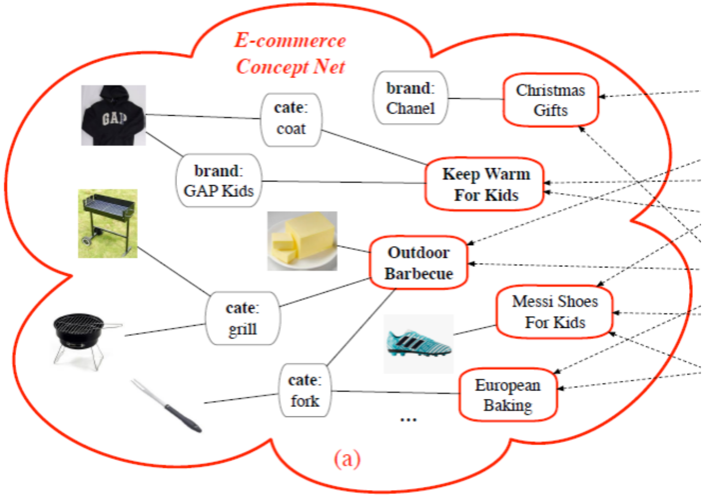
FC: Fully connected

Path Embedding

User-Item-Concept (UIC)
User-Item-Cate-Concpet (UITC)
User-Item-Brand-Concept (UIBC)
User-Cate-Concept (UTC)
User-Brand-Concept (UBC)

How do we know which Item-Concept relations should we choose?

Ans: by **tf-idf score**



Importance of each item-concept (tf-idf)

tf-idf

tf: How many the same description as item i in a concept
idf: How many concepts does item i appear in

$$\mathrm{tf_{i,j}} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

For example:
Concept : Tools for baking
item : spoon

$$\mathrm{idf_i} = \lg \frac{|D|}{|\{j : t_i \in d_j\}|}$$

Concept: include totally 100 items
Same desciption as spoon: 10 items
tf: 10/100 = 0.1

Total Concept: 1000 kinds of concepts Spoon appears in 10 kinds of concepts
000/10) = 2

tf-idf = 0.1* 2 = 0.2

Path Embedding (Take UITC for example)

User Historical Behavior

Category id embedding

$$pi_{\mathrm{UITC}} = \mathrm{CNN}([u_b, i, cate, c_i])$$

Item description embedding

Concept id embedding

$$p_{\mathrm{UITC}} = \mathrm{MaxPooling}(\{pi_{\mathrm{UITC}}\})$$

Average pooling

$$p = f_p(p) = f_p(p_{\mathrm{UIC}}, p_{\mathrm{UITC}}, \cdots)$$

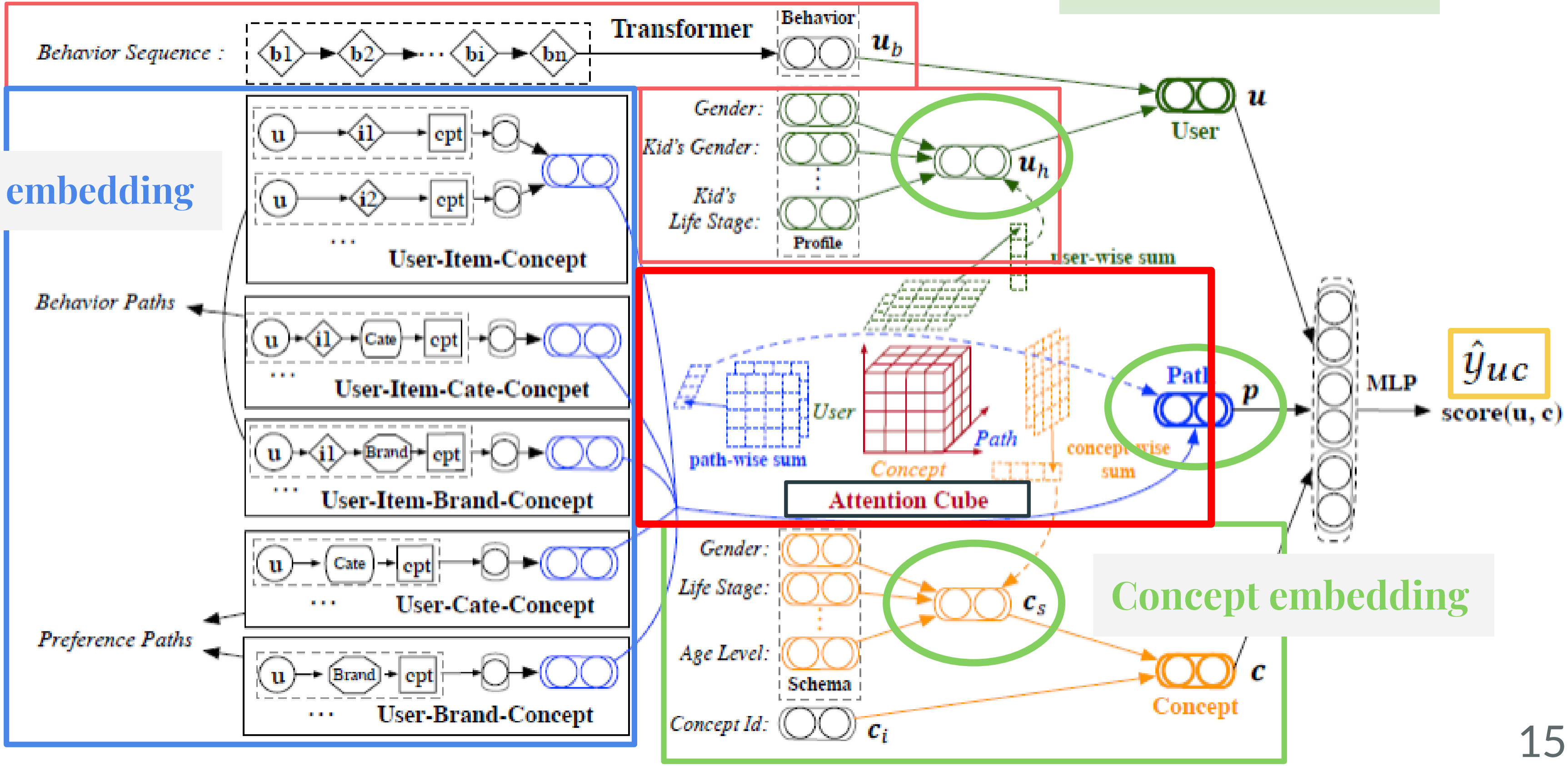
Architecture

User embedding

Average pooling (p11,p12,p15)

Path embedding

Concept embedding



Attention Cube

j-th embedding of meta-path embedding list

$$att_{i,j,k} = {u_h}_i^T W_1 p_j + p_j^T W_2 {c_s}_k + {u_h}_i^T W_3 {c_s}_k$$

i-th embedding of
user profile embedding list

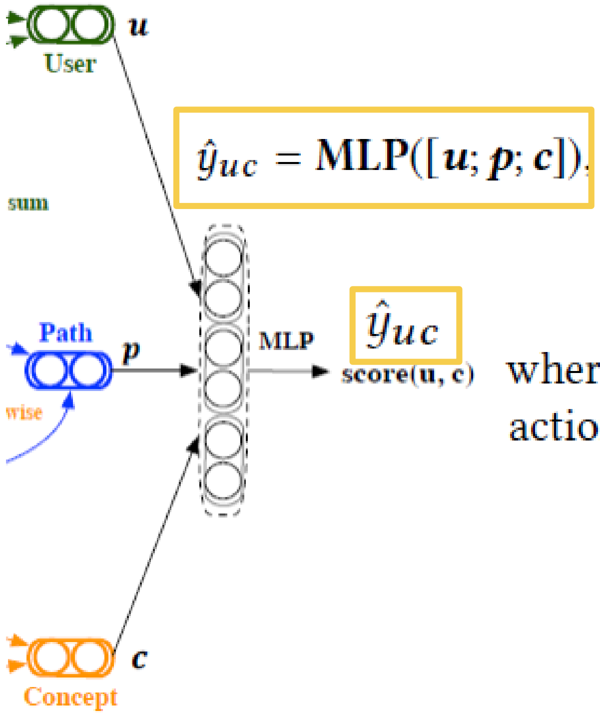
k-th embedding of
concept schema embedding list

$$\boxed{\alpha_{u\,i} = \frac{\exp(\sum_j \sum_k att_{i,j,k})}{\sum_i exp(\sum_j \sum_k att_{i,j,k})}}$$

get $\alpha_{p\,j}$ and $\alpha_{c\,k}$ in a similar way

$$u_h = f_u(u_h) = \alpha_{u\,1} h_{\text{gender}} + \alpha_{u\,2} h_{\text{age}} + \cdots$$

Output & loss function



$$\mathcal{L} = -\sum_{(u,c)\in D^+} \log \hat{y}_{uc} + \sum_{(u,c)\in D^-} \log(1 - \hat{y}_{uc}) \qquad (11)$$

where D^+ and D^- are the positive and negative user-concept interaction pairs.

Outline

- Introduction
- Method
- Experiment
- Conclusion

Dataset

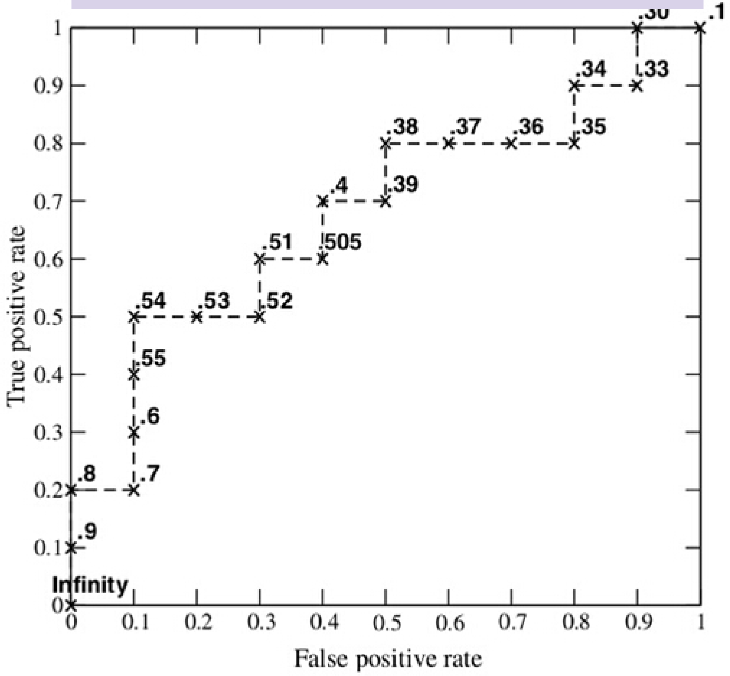
During January 11 to January 14, 2019

	Training	Validation	Testing
# of samples	32,496,827	328,251	1,237,506
# of users	16,120,600	323,544	1,121,475
# of concepts	4,760	2,935	3,176
# of items	438M	76M	141M
# of categories	15,257	11,799	14,590
# of brands	1,434,659	428,036	1,088,480

Table 2: Statistics of Taobao's dataset.

Experiment Method

AUC: Area under ROC curve



HR@k: Hit Ratio

$$HR@K = \frac{NumberOfHits@K}{GT}$$

NDCG@k: Normalized Discounted Cumulative Gain

$$DCG_k = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{\log_2(i+1)}.$$

Relation

$$NDCG_k = \frac{DCG_k}{IDCG_k}$$

Experiment

Meta path with attention → MCRec+

Knowledge graph based → KPRN+

Model	AUC
BPR	0.6005
Wide&Deep	0.6137
MCRec+	0.6447
KPRN+	0.6417
Ours (- att. cube)	0.6403
Ours (full)	**0.6612**

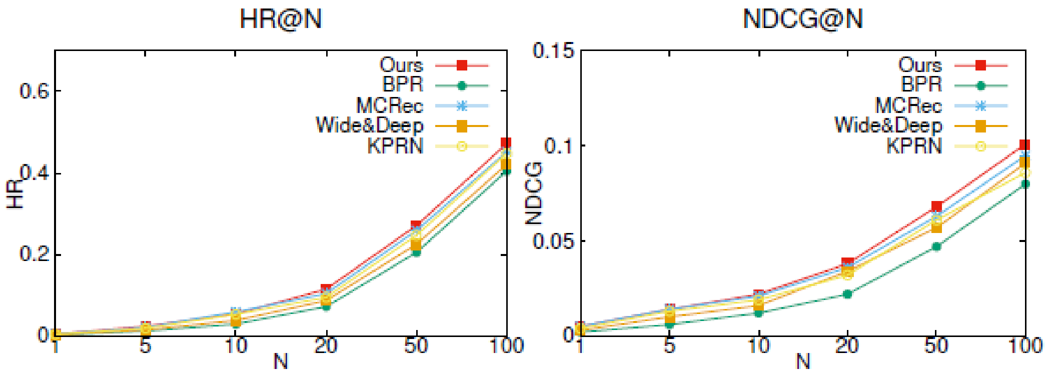


Figure 5: HR and NDCG in Top-N recommendation.

Experiment

Variation	AUC	Decrease (%)
- behavior paths	0.6826	4.03
- preference paths	0.6934	2.41
- all paths	0.6694	6.08
- user behavior sequence	0.7010	1.30
- user profile	0.6986	1.65
- concept schema	0.7031	1.00
Full	**0.7101**	0.0

Table 4: Ablation tests on validation set.

Experiment

Strategy	CTR	Discovery
Rule-based	-	-
MCRec+	+5.1%	+3.4%
Ours	**+6.0%**	**+5.6%**

$$\textbf{Discovery} = \mathrm{Avg}_u\left(\frac{\#\ \text{new clk-cates in 15d}}{\#\ \text{clk-cates}}\right)$$

Conclusion

- We formally define user needs in e-commerce and introduce "e-commerce concept net", where "concepts" can explicitly express various shopping needs for users.
- Based on the e-commerce concept net, we propose a path based deep model with attention cube to infer user needs.
- We evaluate our model outperforms several strong baselines, indicates the value of such user needs inference.