DGRec: Graph Neural Network for Recommendation with Diversified Embedding Generation

Liangwei Yang*
University of Illinois at Chicago
Chicago, USA
lyang84@uic.edu

Shengjie Wang
ByteDance Inc.
Seattle, USA
shengjie.wang@bytedance.com

Yunzhe Tao
ByteDance Inc.
Seattle, USA
yunzhe.tao@bytedance.com

Jiankai Sun
ByteDance Inc.
Seattle, USA
jiankai.sun@bytedance.com

Xiaolong Liu, Philip S. Yu
University of Illinois at Chicago
Chicago, USA
{xliu262,psyu}@uic.edu

Taiqing Wang
ByteDance Inc.
Seattle, USA
taiqing.wang@bytedance.com

WSDM 2023
Code: https://github.com/YangLiangwei/DGRec.

Reported by liang li

Motivation

Motivation：

- The redundancy of the aggregated neighbors and resulting in poor diversity of the recommended list.
- Trapping users in a small subset of familiar items without exploring the vast majority of others.

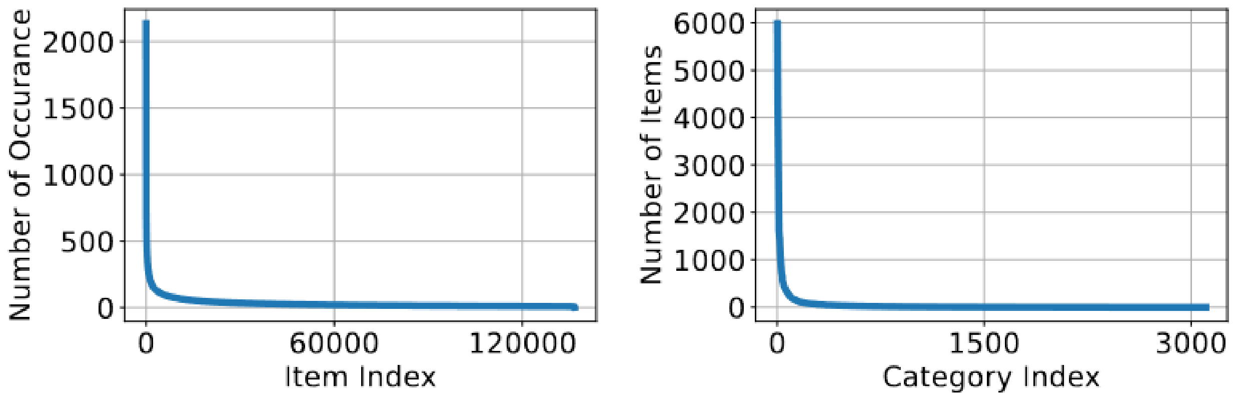
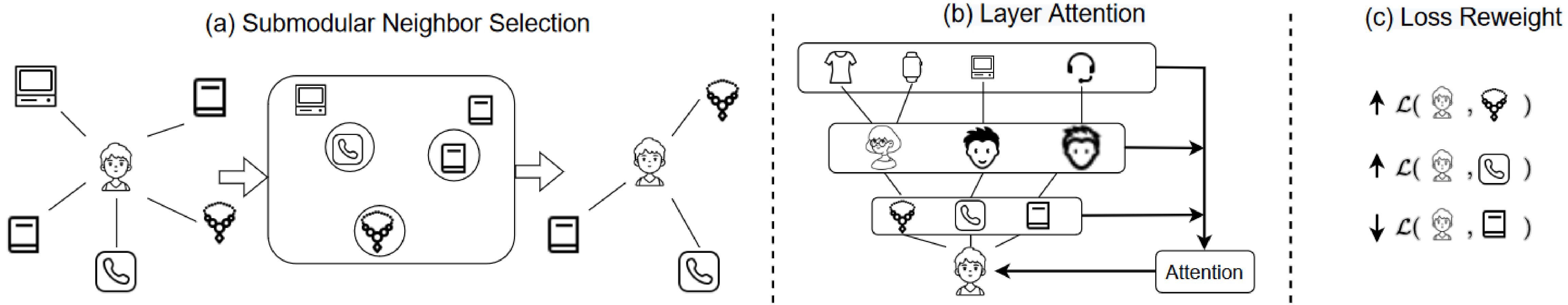


Figure 1: Long tail distribution in Recommender System on TaoBao dataset [49].

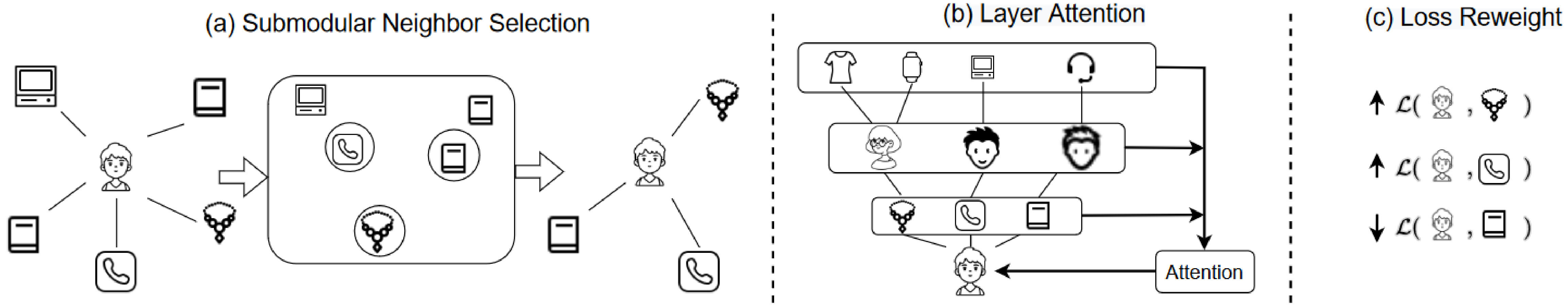
Problem Statement



$\mathcal{U} = \{u_1, u_2, ..., u_{|\mathcal{U}|}\}$ $C(\cdot)$ that maps each item to its category

$\mathcal{I} = \{i_1, i_2, ..., i_{|\mathcal{I}|}\}$ $\mathbf{R} \in \mathbb{R}^{|\mathcal{U}|\times|\mathcal{I}|}$ $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ $\mathcal{V} = \mathcal{U} \cup \mathcal{I}$

Method



$$\mathbf{e}_u^{(l+1)} = \mathbf{e}_u^{(l)} \oplus \mathrm{AGG}^{(l+1)}(\{\mathbf{e}_i^{(l)} \mid i \in \mathcal{N}_u\}), \quad (1)$$

$$f(v|A) \geq f(v|B) \quad \forall A \subset B \subset V,\ v \in V \text{ and } v \notin B. \quad (2)$$

$$f(v|A) := f(\{v\} \cup A) - f(A)$$

对所有$X, Y \subseteq \Omega$,其中$X \subseteq Y$,则对所有$x \in \Omega \setminus Y$,有

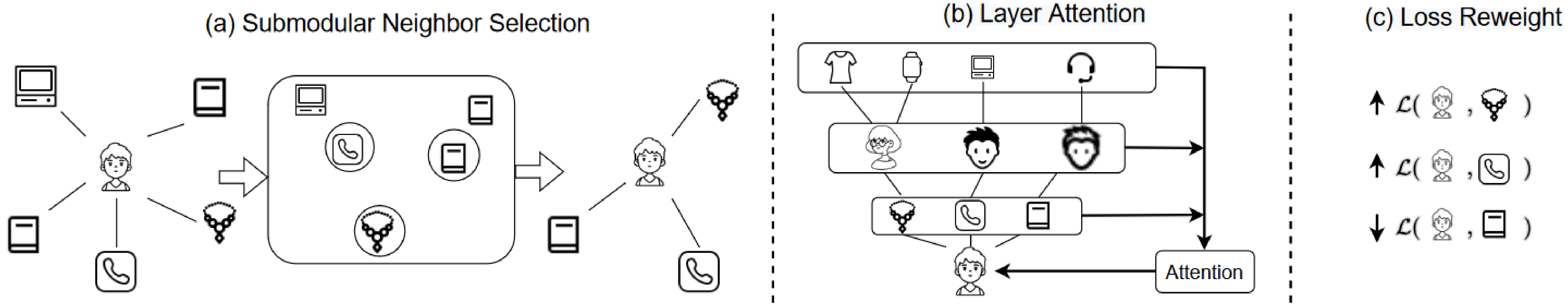
$$f(X \cup \{x\}) - f(X) \geqslant f(Y \cup \{x\}) - f(Y)$$

$$\mathbf{E}^{(0)} = \left(\mathbf{e}_1^{(0)}, \mathbf{e}_2^{(0)}, \ldots, \mathbf{e}_{|\mathcal{U}|+|\mathcal{I}|}^{(0)}\right), \quad (3)$$

$$\mathbf{e}_u^{(l+1)} = \sum_{i \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u|}\sqrt{|\mathcal{N}_i|}} \mathbf{e}_i^{(l)},$$

$$\mathbf{e}_i^{(l+1)} = \sum_{u \in \mathcal{N}_i} \frac{1}{\sqrt{|\mathcal{N}_i|}\sqrt{|\mathcal{N}_u|}} \mathbf{e}_u^{(l)}, \quad (4)$$

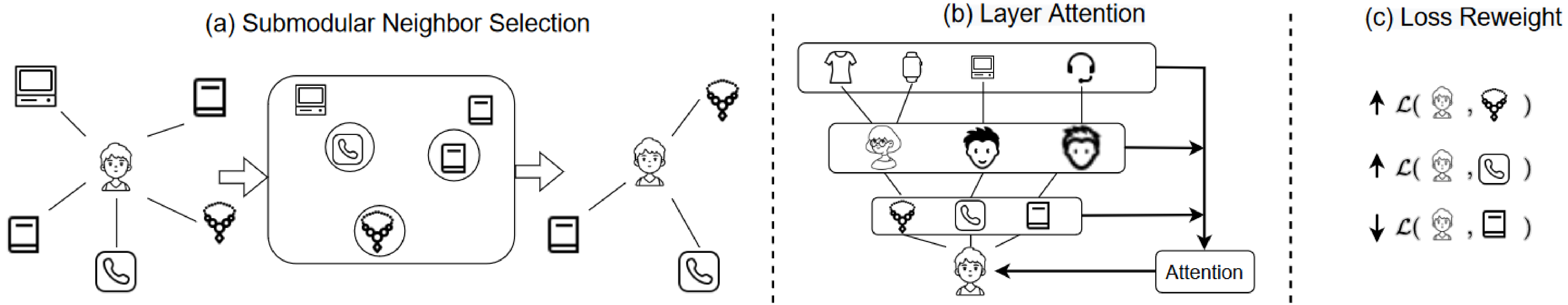
Method



$$\mathbf{e}_u = \text{Layer\_Attention}\left(\mathbf{e}_u^{(0)}, \mathbf{e}_u^{(1)}, \ldots, \mathbf{e}_u^{(\text{layer num})}\right),$$
$$\mathbf{e}_i = \text{Layer\_Attention}\left(\mathbf{e}_i^{(0)}, \mathbf{e}_i^{(1)}, \ldots, \mathbf{e}_i^{(\text{layer num})}\right). \quad (5)$$

$$\mathcal{L} = \sum_{(u,i)\in\mathcal{E}} w_{C(i)} \mathcal{L}_{bpr}(u, i, j) + \lambda\|\Theta\|_2^2, \quad (6)$$

Method



$$f(\mathcal{S}_u) = \sum_{i \in \mathcal{N}_u \backslash \mathcal{S}_u} \max_{i' \in \mathcal{S}_u} \mathrm{sim}(i, i'), \tag{7}$$

$$\mathrm{sim}(i, i') = \exp\left(-\frac{||\mathbf{e}_i - \mathbf{e}_{i'}||^2}{\sigma^2}\right). \tag{8}$$

$$\mathcal{S}_u \leftarrow \mathcal{S}_u \cup i^*,$$
$$i^* = \arg\max_{i \in \mathcal{N}_u \backslash \mathcal{S}_u} [f(\mathcal{S}_u \cup i) - f(\mathcal{S}_u)]. \tag{9}$$

$$\mathbf{e} = \mathrm{Readout}([\mathbf{e}^{(0)}, \mathbf{e}^{(1)}, \ldots, \mathbf{e}^{(L)}]) = \sum_{l=0}^{L} a^{(l)} \mathbf{e}^{(l)}, \tag{10}$$

$$a^{(l)} = \frac{\exp(\langle \mathbf{W}_{\mathrm{Att}}, \mathbf{e}^{(l)} \rangle)}{\sum_{l'=0}^{\mathrm{L}} \exp(\langle \mathbf{W}_{\mathrm{Att}}, \mathbf{e}^{(l')} \rangle)}. \tag{11}$$

$$w_{C(i)} = \frac{1 - \beta}{1 - \beta^{|C(i)|}}, \tag{12}$$

Experiments

Table 1: Statistics of the Datasets

Dataset	TaoBao	Beauty
Users	82,633	8,159
Items	136,710	5,862
Interactions	4,230,631	98,566
Categories	3,108	41
Average Category Size	43.986	139.595

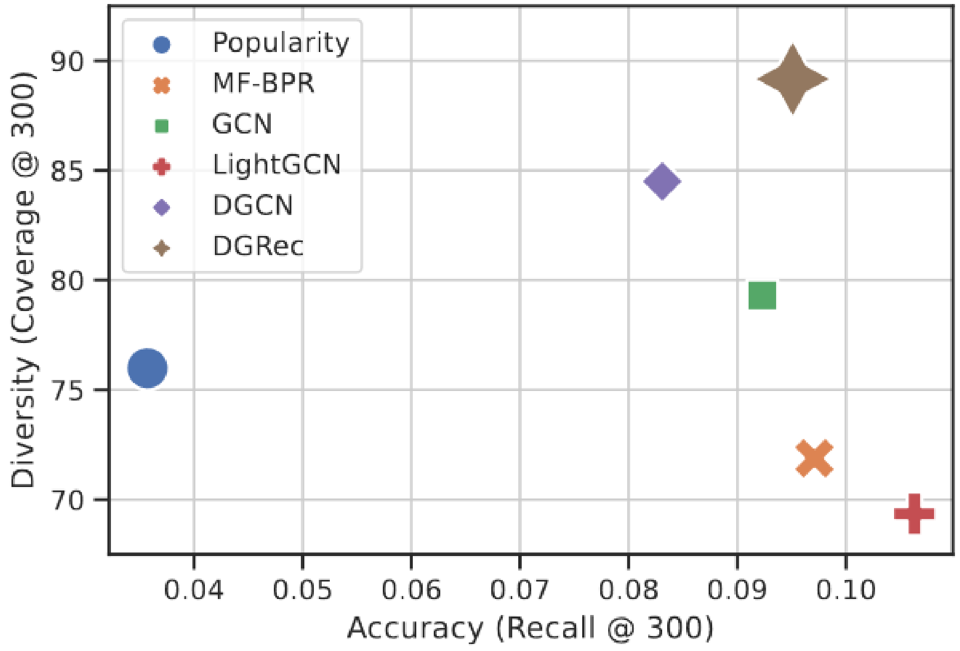


Figure 3: Accuracy-Diversity trade-off comparison on TaoBao dataset. The upper-right model is enlarged.

Experiments

Table 2: Overall comparison on TaoBao dataset, the best and second-best results are in bold and underlined, respectively.

Method	TaoBao					
	Recall@100	Recall@300	HR@100	HR@300	Coverage@100	Coverage@300
Popularity	0.0186	0.0357	0.1496	0.2562	<u>38.2449</u>	75.9837
MF-BPR [29]	<u>0.0487</u>	<u>0.0971</u>	<u>0.3103</u>	<u>0.4889</u>	34.0812	71.8802
GCN [18]	0.0446	0.0923	0.2840	0.4634	37.2577	79.2985
LightGCN [14]	**0.0528**	**0.1063**	**0.3261**	**0.5097**	32.7069	69.3502
DGCN [49]	0.0394	0.0831	0.2634	0.4369	38.1183	<u>84.4989</u>
DGRec	0.0472	0.0951	0.3026	0.4817	**39.0597**	**89.1684**

Table 3: Overall comparison on Beauty dataset, the best and second-best results are in bold and underlined, respectively.

Method	Beauty					
	Recall@100	Recall@300	HR@100	HR@300	Coverage@100	Coverage@300
Popularity	0.1012	0.2096	0.1833	0.3124	16.0213	**27.9336**
MF-BPR [29]	0.2310	0.3863	0.3404	0.4966	15.8728	25.6659
GCN [18]	0.2388	0.3897	<u>0.3423</u>	0.3897	16.5311	25.5634
LightGCN [14]	**0.2517**	**0.4205**	**0.3688**	**0.5318**	15.0203	23.9421
DGCN [49]	0.2395	0.3790	0.3418	0.4792	<u>18.2876</u>	26.9694
DGRec	<u>0.2399</u>	<u>0.3915</u>	0.3420	<u>0.5021</u>	**19.0557**	<u>27.5704</u>

Experiments

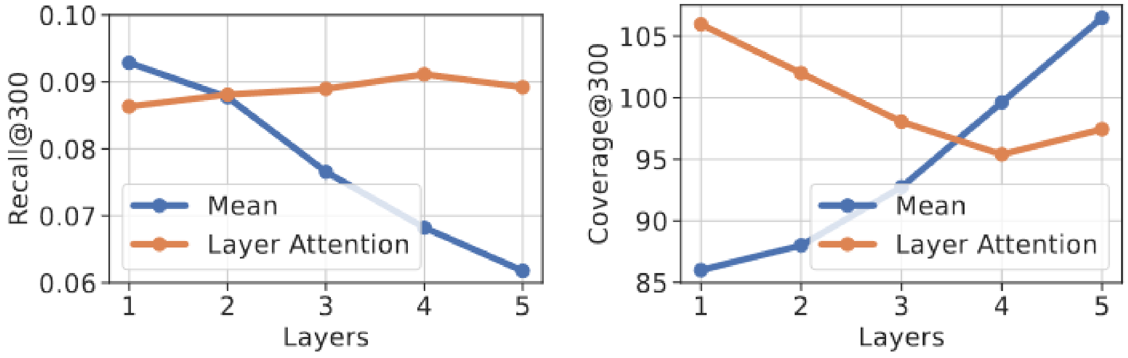


Figure 4: Layer combination experiments on TaoBao dataset. Mean combines embedding learned from different layers by mean average. Layer attention combines these embeddings by the attention module illustrated in Section 3.3.

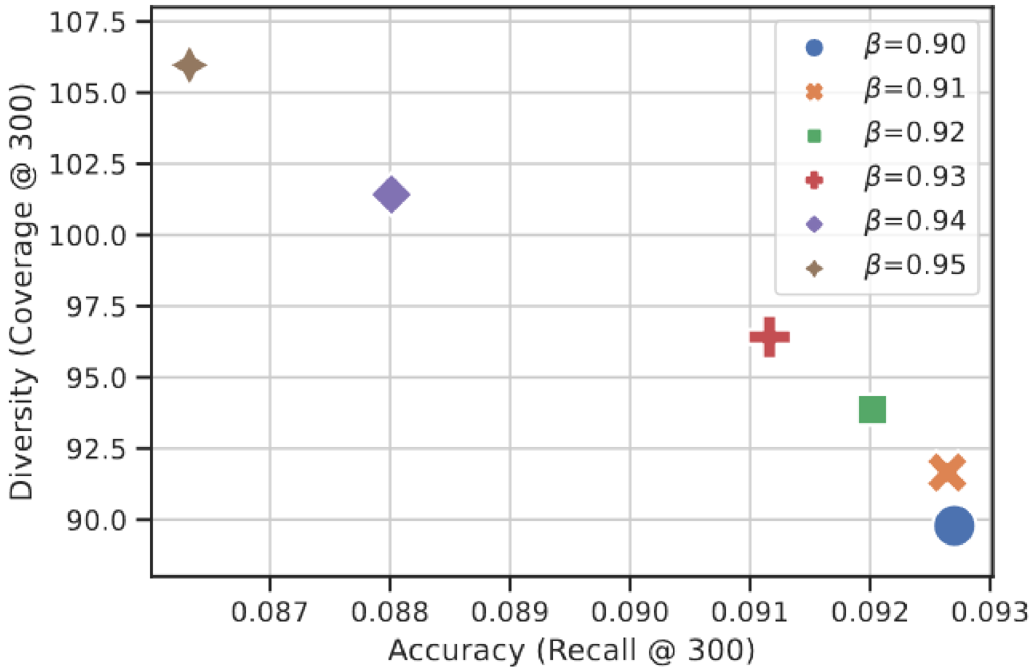


Figure 5: Accuracy-diversity trade-off by loss reweighting.

Experiments

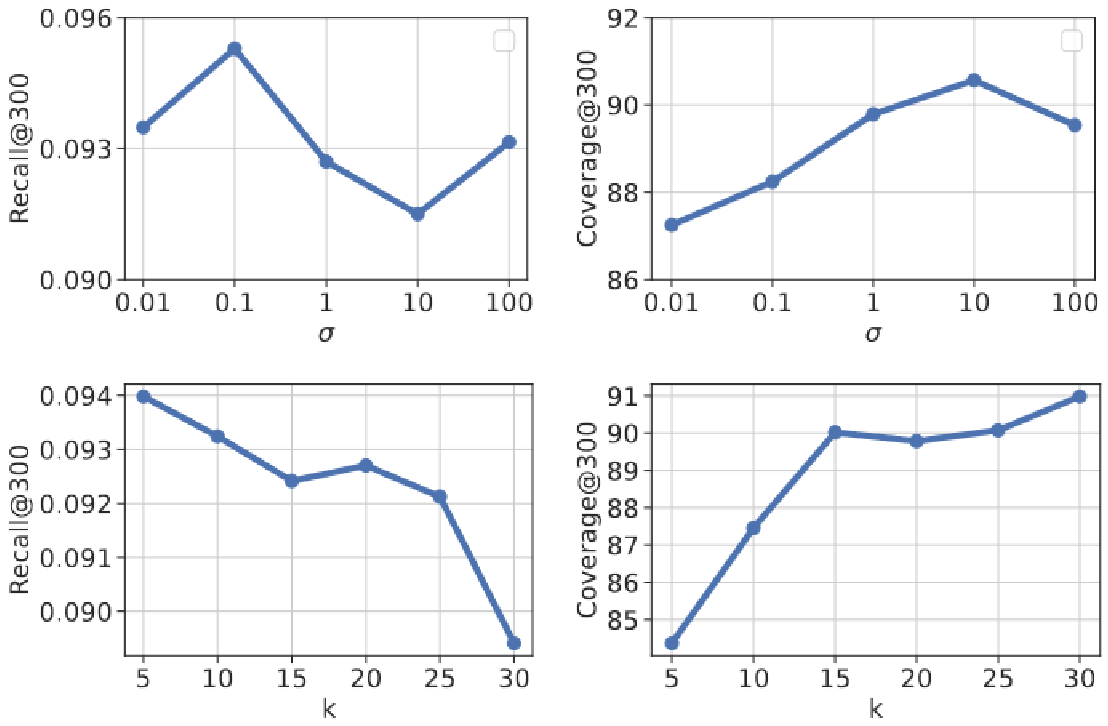


Figure 6: Parameter sensitivity of σ and k. σ controls the similarity computation and k is the number of selected neighbors.

Experiments

Table 4: Ablation study on TaoBao dataset. We show DGRec' performance when removing each of the modules.

Method	R@300	HR@300	C@300
DGRec	0.0951	0.4817	89.1684
w/o Submodular selection	0.0982	0.4869	84.9129
w/o Layer attention	0.1009	0.4976	82.9553
w/o Loss reweighting	0.0886	0.4612	79.3286

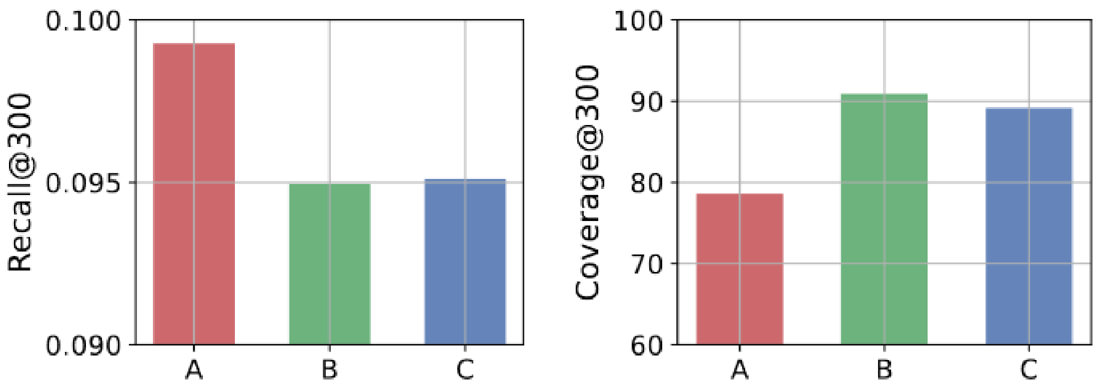


Figure 7: The influence of different submodular functions. A: bucket coverage function, B: category coverage function, and C: facility location function.

Thanks