Negative Can Be Positive: Signed Graph Neural Networks for Recommendation

Junjie Huang [1,2] Ruobing Xie[3] Qi Cao[1,2] Huawei Shen[1,2] Shaoliang Zhang[3] Feng Xia[3] Xueqi Cheng[1,2]

[1]Institute of Computing Technology, Chinese Academy of Sciences [2]University of Chinese Academy of Sciences
[3]WeChat, Tencent

Abstract

Most of the existing GNN-based recommender system models focus on learning users' personalized preferences from these (explicit/implicit) positive feedback to achieve personalized recommendations. However, in the real-world recommender system, the users' feedback behavior also includes negative feedback behavior (e.g., click *dislike* button), which also reflects users' personalized preferences. **How to utilize negative feedback is a challenging research problem.** In this paper, we first qualitatively and quantitatively analyze the three kinds of negative feedback that widely existed in real-world recommender systems and investigate the role of negative feedback in recommender systems. We found that it is different from what we expected – **not all negative items are ranked low, and some negative items are even ranked high in the overall items.** Then, we propose a novel Signed Graph Neural Network Recommendation model (SiGRec) to encode the users' negative feedback behavior. Our SiGRec can learn positive and negative embeddings of users and items via positive and negative graph neural network encoders, respectively. Besides, we also define a new Sign Cosine (SiC) loss function to adaptively mine the information of negative feedback for different types of negative feedback. **Extensive experiments on four datasets demonstrate the proposed model outperforms several existing models.** Specifically, on the Zhihu dataset, SiGRec outperforms the unsigned GNN model (i.e. LightGCN), 12.86%, 12.63%, and 12.28% in P@20, R@20, and nDCG@20, respectively. We hope our work can open the door to further exploring the negative feedback in recommendations.

Introduction

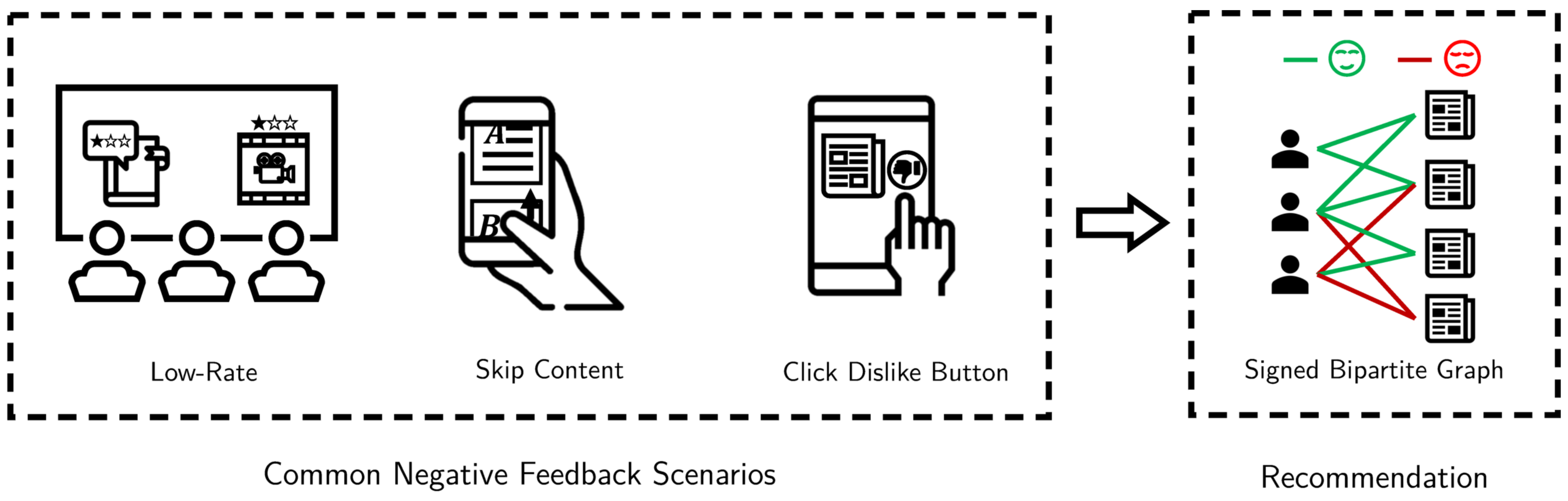

Figure. 1. The illustration of common negative feedback interactions in recommender systems, including low rating, skipping recommended content and clicking *dislike* button. These scenarios with both positive and negative interactions between users and items can be modeled as a signed bipartite graph for better recommendation.

Bipartite Graph in Recommendation: A bipartite graph is denoted as $\mathcal{G} = (\mathcal{U}, \mathcal{I}, \mathcal{E})$, where $\mathcal{U}$ denotes the user set, $\mathcal{I}$ denotes the item set, and $\mathcal{E}$ is the set of weighted edges between $\mathcal{U}$ and $\mathcal{I}$.

Message-passing Scheme:

$$\begin{aligned} z_w^j &= f_{\text{aggregate}}\left(\{z_v^{j-1} \mid v \in \mathcal{N}^+{}_w \cup \{w\}\}\right) \\ z_w &= f_{\text{update}}\left([z_w^0, z_w^1, \ldots, z_w^l]\right) \end{aligned} \tag{1}$$

Negative Edge in Recommendation: Given a bipartite graph $\mathcal{G} = (\mathcal{U}, \mathcal{I}, \mathcal{E})$, we define the edge is negative $(-)$ when the value of the edge is less than the threshold τ, the other edges are defined as positive $(+)$.

Signed Bipartite Graph in Recommendation: After the Negative Edge in Recommendation definition, the bipartite graph can be transformed to a signed bipartite graph $\mathcal{G} = \mathcal{G}^+ \bigcup \mathcal{G}^- = (\mathcal{U}, \mathcal{I}, \mathcal{E}^+, \mathcal{E}^-)$, where $\mathcal{U}$ and $\mathcal{I}$ are the two sets of nodes like before, $\mathcal{E}^+$ and $\mathcal{E}^-$ denote the positive edge set and negative edge set. $\mathcal{E} = \mathcal{E}^+ \bigcup \mathcal{E}^-$ and $\mathcal{E}^+ \bigcap \mathcal{E}^- = \emptyset$ means that the sign for an edge e_{ui} can be only postive or negative. Similarly, $\mathcal{A}^+ \in \mathbb{R}^{(|\mathcal{U}|+|\mathcal{I}|)\times(|\mathcal{U}|+|\mathcal{I}|)}$ and $\mathcal{A}^- \in \mathbb{R}^{(|\mathcal{U}|+|\mathcal{I}|)\times(|\mathcal{U}|+|\mathcal{I}|)}$ are denoted as the adjacency matrix for positive bipartite graph $\mathcal{G}^+$ and negative bipartite graph $\mathcal{G}^-$, respectively.

Contributions

- We qualitatively and quantitatively investigated the role of negative feedback in recommender systems. As far as we know, we are the first to systematically analyze the effect of three different types of negative feedback for recommender systems. We found that negative feedback can be positive for item ranking and model training.
- Based on our findings on negative items, we introduced a new Signed Graph Neural Network Recommendation model (SiGRec). Our model can adaptively model various negative feedbacks and learn effective embeddings for recommendation tasks.
- We conducted recommendation experiments on four real-world datasets with negative feedback. Our methods achieved state-of-the-art performances, compared with MF-based methods, unsigned GNN-based methods, and signed GNN-based methods.

Analysis and Motivation on Negative Feedback

Table 1. Statistics of four real-world datasets used in this paper.

Dataset	Amazon-Book	Yelp	Zhihu	WeChat
# Users	35,736	38,595	7,860	8,582
# Items	38,121	27,823	9,577	4,975
# Interactions	1,960,674	1,900,308	554,150	274,375
% Density	0.14	0.18	0.74	0.67
% Negative Ratio	19.4	31.8	70.7	14.3
Scenarios	Low Rate		Skip	Click *Dislike* Button

- We can find that negative items show some positive effects on item ranking and model training.
- Besides, among these three scenarios, clicking dislike button shows the most negative interest.
- These findings can be due to skipping contents is relatively frequent and low-cost, so the negative emotions are not strong; although low rating has direct negative feedback from users, this behavior is after the user has watched/purchased the product (It means that the users at least have positive interest in the early stage).
- Clicking dislike button means more negative emotions on recommended items.

Methodology

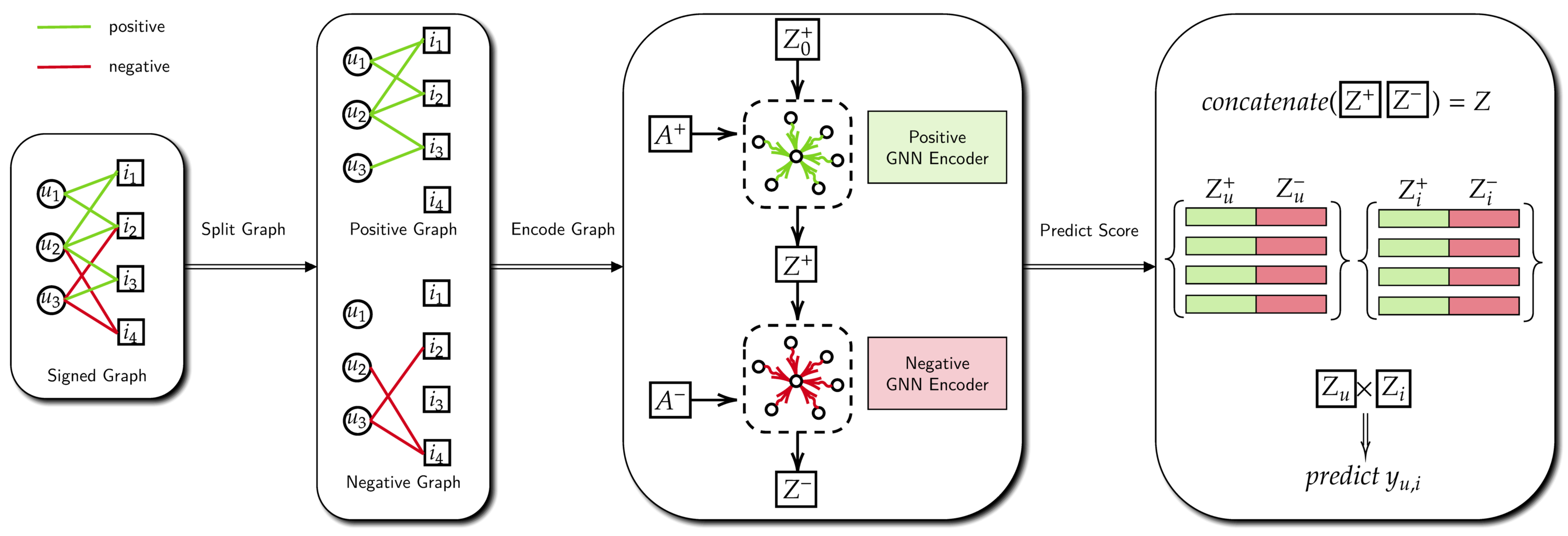

Figure. 2. The demonstration of our SiGRec. SiGRec includes a positive GNN encoder f^+ and a negative GNN encoder f^-. The final embeddings are the concatenation of positive embeddings and negative embeddings.

Loss Function

Bayesian Personalized Ranking Loss Function:

$$\mathcal{L}_{bpr} = \sum_{(u,i_1,i_2)\in O} -\log\sigma\left(\hat{y}_{u,i_1} - \hat{y}_{u,i_2}\right), \tag{2}$$

Sign Cosine Loss Function:

$$\mathcal{L}_{sign}(z_u, z_i, y_{u,i}) = \begin{cases} 1 - \cos(z_u, z_i), & \text{if } y_{u,i} = 1 \\ \omega \cdot \max(0, \cos(z_u, z_i) - \mu), & \text{if } y_{u,i} = -1 \end{cases}, \tag{3}$$

Experiments

Table 2. Performance comparison of different models on four datasets. The performances are the average of five experiments from different seed sets. The best method is **bold**, and the second best is <u>underlined</u>. * is the significant level of 0.05 for the t-test with the best baselines.

Dataset	Metric	BPRMF	NeuMF	NGCF	DGCF	LightGCN	SiReN	SiGRec	Improv(%)	Power(%)
Amazon-Book	P@10	0.0427	0.0314	0.0415	0.0496	<u>0.0549</u>	0.0536	**0.0565***	2.77	99.9%
	R@10	0.0586	0.0427	0.0560	0.0672	<u>0.0744</u>	0.0741	**0.0764***	2.66	95.6%
	nDCG@10	0.0614	0.0423	0.0588	0.0715	<u>0.0797</u>	0.0773	**0.0815***	2.16	86.2%
	P@15	0.0380	0.0289	0.0372	0.0437	<u>0.0484</u>	0.0476	**0.0499***	3.27	100%
	R@15	0.0773	0.0580	0.0746	0.0876	0.0968	<u>0.0972</u>	**0.0999***	2.84	84.9%
	nDCG@15	0.0672	0.0476	0.0647	0.0777	<u>0.0864</u>	0.0846	**0.0885***	2.50	95.9%
	P@20	0.0347	0.0269	0.0342	0.0398	<u>0.0438</u>	0.0434	**0.0454***	3.70	100%
	R@20	0.0930	0.0712	0.0906	0.1051	0.1158	<u>0.1169</u>	**0.1199***	2.56	89.3%
	nDCG@20	0.0726	0.0523	0.0702	0.0836	<u>0.0926</u>	0.0913	**0.0952***	2.76	99.1%
Yelp	P@10	0.0287	0.0210	0.0298	0.0335	0.0358	<u>0.0367</u>	**0.0402***	9.57	100%
	R@10	0.0449	0.0331	0.0462	0.0524	0.0557	<u>0.0579</u>	**0.0626***	8.03	100%
	nDCG@10	0.0427	0.0305	0.0441	0.0502	0.0535	<u>0.0555</u>	**0.0602***	8.49	100%
	P@15	0.0261	0.0196	0.0269	0.0302	0.0321	<u>0.0328</u>	**0.0360***	9.75	100%
	R@15	0.0610	0.0458	0.0624	0.0703	0.0745	<u>0.0772</u>	**0.0836***	8.39	100%
	nDCG@15	0.0482	0.0351	0.0496	0.0562	0.0597	<u>0.0618</u>	**0.0671***	8.50	100%
	P@20	0.0242	0.0185	0.0251	0.0278	0.0296	<u>0.0303</u>	**0.0331***	9.36	100%
	R@20	0.0749	0.0575	0.0769	0.0859	0.0913	<u>0.0945</u>	**0.1020***	7.88	100%
	nDCG@20	0.0531	0.0392	0.0547	0.0617	0.0656	<u>0.0679</u>	**0.0735***	8.23	100%
Zhihu	P@10	0.0237	0.0213	0.0276	0.0279	0.0301	<u>0.0368</u>	**0.0392***	6.40	99.8%
	R@10	0.0394	0.0356	0.0464	0.0467	0.0506	<u>0.0637</u>	**0.0681***	7.00	100%
	nDCG@10	0.0348	0.0315	0.0413	0.0417	0.0451	<u>0.0556</u>	**0.0603***	8.46	100%
	P@15	0.0216	0.0196	0.0257	0.0254	0.0276	<u>0.0336</u>	**0.0355***	5.69	99.9%
	R@15	0.0536	0.0488	0.0641	0.0633	0.0694	<u>0.0862</u>	**0.0913***	5.95	100%
	nDCG@15	0.0402	0.0365	0.0481	0.0479	0.0523	<u>0.0642</u>	**0.0691***	7.60	100%
	P@20	0.0203	0.0184	0.0240	0.0236	0.0257	<u>0.0313</u>	**0.0328***	5.09	99.8%
	R@20	0.0668	0.0607	0.0798	0.0778	0.0859	<u>0.1067</u>	**0.1115***	4.50	99.8%
	nDCG@20	0.0454	0.0412	0.0542	0.0537	0.0588	<u>0.0722</u>	**0.0771***	6.78	99.9%
WeChat	P@10	0.0516	0.0469	0.0530	0.0549	<u>0.0562</u>	0.0474	**0.0582***	3.41	79.6%
	R@10	0.0882	0.0804	0.0912	0.0943	<u>0.0970</u>	0.0821	**0.1001***	3.20	63.6%
	nDCG@10	0.0786	0.0708	0.0808	0.0843	<u>0.0871</u>	0.0716	**0.0890**	2.13	28.9%
	P@15	0.0463	0.0429	0.0477	0.0494	<u>0.0507</u>	0.0430	**0.0518**	2.13	51.7%
	R@15	0.1184	0.1096	0.1221	0.1271	<u>0.1306</u>	0.1109	**0.1353**	1.85	25.2%
	nDCG@15	0.0910	0.0829	0.0936	0.0978	<u>0.1011</u>	0.0836	**0.1044**	1.52	19.3%
	P@20	0.0425	0.0397	0.0440	0.0452	<u>0.0467</u>	0.0401	**0.0479**	2.14	72.9%
	R@20	0.1440	0.1346	0.1494	0.1536	<u>0.1591</u>	0.1374	**0.1640**	2.17	50.1%
	nDCG@20	0.1014	0.0930	0.1046	0.1086	<u>0.1126</u>	0.0943	**0.1159**	1.67	31.6%