SIGformer: Sign-aware Graph Transformer for Recommendation

Sirui Chen
Zhejiang University
The State Key Laboratory of
Blockchain and Data Security
Hangzhou, China
chenthree@zju.edu.cn

Jiawei Chen*
Zhejiang University
The State Key Laboratory of
Blockchain and Data Security
Hangzhou, China
sleepyhunt@zju.edu.cn

Sheng Zhou
Zhejiang University
Hangzhou, China
zhousheng_zju@zju.edu.cn

Bohao Wang
Zhejiang University
The State Key Laboratory of
Blockchain and Data Security
Hangzhou, China
bohao.wang@zju.edu.cn

Shen Han
Huazhong Agricultural University
Wuhan, China
hanshen@webmail.hzau.edu.cn

Chanfei Su
OPPO Co Ltd
Shenzhen, China
suchanfei@oppo.com

Yuqing Yuan
OPPO Co Ltd
Shenzhen, China
yuanyuqing@oppo.com

Can Wang
Zhejiang University
The State Key Laboratory of
Blockchain and Data Security
Hangzhou, China
wcan@zju.edu.cn

)分别处理正负反馈，无法全面利用签名图中的协同信息;2)他们依赖mlp或gnn从负反馈中提取信息，这可能并不有效。

ABSTRACT

In recommender systems, most graph-based methods focus on positive user feedback, while overlooking the valuable negative feedback. Integrating both positive and negative feedback to form a signed graph can lead to a more comprehensive understanding of user preferences. However, the existing efforts to incorporate both types of feedback are sparse and face two main limitations: 1) They process positive and negative feedback separately, which fails to holistically leverage the collaborative information within the signed graph; 2) They rely on MLPs or GNNs for information extraction from negative feedback, which may not be effective.

To overcome these limitations, we introduce *SIGformer*, a new method that employs the transformer architecture to sign-aware graph-based recommendation. SIGformer incorporates two innovative positional encodings that capture the spectral properties and path patterns of the signed graph, enabling the full exploitation of the entire graph. Our extensive experiments across five real-world datasets demonstrate the superiority of SIGformer over state-of-the-art methods. The code is available at https://github.com/StupidThree/SIGformer.

*Corresponding author.

SIGIR '24, July 14–18, 2024, Washington, DC, USA

ACM ISBN 979-8-4007-0431-4/24/07
https://doi.org/10.1145/3626772.3657747

CCS CONCEPTS

• **Information systems** → **Recommender systems**.

KEYWORDS

Sign-aware Recommendation, Graph, Transformer

ACM Reference Format:
Sirui Chen, Jiawei Chen, Sheng Zhou, Bohao Wang, Shen Han, Chanfei Su, Yuqing Yuan, and Can Wang. 2024. SIGformer: Sign-aware Graph Transformer for Recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '24), July 14–18, 2024, Washington, DC, USA*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3626772.3657747

1 INTRODUCTION

Recent years have witnessed a surge of graph-based methods for recommendation [20, 24, 37, 54, 66]. These methods generally start by creating a bipartite graph from users' historical feedback and then employ graph-enhanced representation techniques (*e.g.*, Graph Neural Network) to learn embeddings for both users and items. Owing to the graph structure's inherent capacity to encapsulate collaborative relations among users and items, graph-based methods have achieved state-of-the-art performance in collaborative recommendation.

However, most existing graph-based methods focus on user positive feedback, while the rich negative feedback is often overlooked. In practice, negative feedback is readily available in many Recommendation Systems (RS) — users can give low ratings, click the "dislike" button, or directly skip items on various platforms like Amazon, Taobao, and TikTok. This crucial negative feedback not only directly indicates users' preferences but also provides valuable collaborative information benefiting recommendation. To illustrate this point, consider Figure 1, where positive and negative feedback

具体来说，我们结合了符号图的拉普拉斯矩阵的低频特征向量作为位置编码。

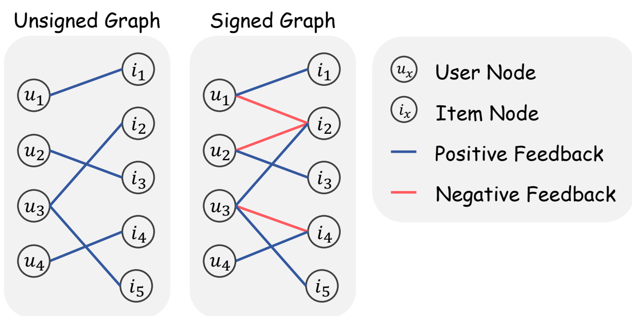

Figure 1: Illustration of sign-aware recommender system, where the signed graph can be constructed from both positive and negative feedback, carrying richer collaborative information.

among users and items are constructed as a signed graph. The high-order connectivity through negative feedback also conveys useful collaborative insights. For instance, path $< u_1 \overset{-}{-} i_2 \overset{-}{-} u_2 >$ suggests users u_1 and u_2 may have similar preferences as they both give negative feedback to item i_2. Additionally, the interplay of positive and negative relations offers richer collaborative relations. For example, path $< u_3 \overset{-}{-} i_4 \overset{+}{-} u_4 >$ highlights the different preferences between u_3 and u_4; a longer path $< u_1 \overset{-}{-} i_2 \overset{-}{-} u_2 \overset{+}{-} i_3 >$ implies that user u_1 is likely to favor i_3 as his similar user u_2 has previously interacted with it.

Acknowledging the valuable collaborative information supplied by negative feedback, the integration of both positive and negative feedback presents a promising direction for enhancing graph-based recommendation. However, to the best of our knowledge, only a few studies have investigated this domain [30, 42, 48]. These methods generally construct two separate graphs from positive and negative feedback and then learn distinct representations from each, subsequently merging these representations for predictions. Despite decent performance, we identify two significant limitations:

以往研究先构建两个图再合并，忽略了协同信息

- **The positive and negative feedback is processed separately, without a holistic consideration.** As previously discussed, the integration of positive and negative feedback within a graph offers rich collaborative information, reflecting the levels of similarity between users and items. Fully exploiting such information warrants the direct utilization of the entire signed graph, rather than processing separate subgraphs independently.
- **The effectiveness of MLPs or GNNs in extracting information from the negative graph is questionable.** Most GNNs, particularly those tailored for recommendation (*e.g.,* LightGCN [24]), are based on the homophily assumption — *i.e.,* connected nodes are likely to be similar. This assumption does not hold for the negative graph. Meanwhile, MLPs struggle to fully utilize the graph structure and are challenging to train effectively in recommendation scenarios due to data sparsity.

gnn对负反馈不成立，数据稀疏mlp难训练

Given the shortcomings of existing methods, we argue for the necessity of a new architecture that can fully exploit the entire signed graph. Inspired by the success of transformer architecture in many fields including language processing [2, 33, 52], computer visions [4, 7, 18] and sequential recommendation [21, 50, 59], we propose leveraging transformer in this scenario. Indeed, transformer is highly aligned with the fundamental principles of collaborative filtering — *i.e.,* estimating similarity between users and items according to their historical feedback, and then aggregating information from those similar entities for predictions. While appealing, adapting transformer to sign-aware graph-based recommendation is non-trivial. The vanilla transformer focuses solely on semantic similarity via self-attention, lacking an explicit encoding of the collaborative information in the signed graph. Although existing graph transformer models [6, 36, 65] introduce subtle positional encodings to capture graph structures, they are neither specifically designed for the signed graph nor the recommendation task. To address these challenges, we introduce two novel positional encodings tailored for sign-aware graph-based recommendation:

(1) Sign-aware Spectral Encoding (SSE). To integrate the structure of the entire signed graph, we propose to utilize the node spectral representation on the signed graph. Specifically, we incorporate the low-frequency eigenvectors of the signed graph's Laplacian matrix as positional encoding. Our theoretical analysis supports the efficacy of this approach: the transformer equipped with SSE can be interpreted as a low-pass filter, bringing the embeddings of user-item pairs with positive feedback closer and distancing those with negative feedback.

使具有正反馈的用户-项目对的嵌入更接近，并使具有负反馈的用户-项目对的嵌入距离更远

(2) Sign-aware Path Encoding (SPE). To further capture collaborative relations among users and items, we focus on the patterns of paths within the signed graph. We encode the distance and the signs of edges along these paths into learnable parameters to capture the affinity between nodes connected by these paths. This design is based on our intuition that different path types reflect varying levels of similarity.

Equipped with these encodings, we introduce a novel recommendation method named **SI**gn-aware Graph Trans**former** (**SIGformer**), which adeptly utilizes the collaborative information within the signed graph. Its effectiveness is validated through empirical experiments on five real-world datasets, where it significantly outperforms existing graph-based methods. Additional ablation studies further confirm the critical role of incorporating negative feedback and the efficacy of our specifically designed encodings.

Our contributions are summarized as follows:

- We highlight the importance of integrating negative feedback in graph-based recommendation and advocate for the application of transformer architecture for sign-aware graph-based recommendation.
- We propose two innovative sign-aware positional encodings, derived from the perspectives of signed graph spectrum and paths, which fully exploit the sign-aware collaborative information.
- We propose SIGformer and conduct extensive experiments to validate the superiority of our SIGformer over state-of-the-art methods.

2 PRELIMINARY

In this section, we present the background of sign-aware graph-based recommendation and transformer model.

2.1 Sign-aware Graph-based Recommendation

Suppose we have a recommender system (RS) with a user set $\mathcal{U}$ and an item set $\mathcal{I}$. Let n and m be the number of users and items in RS. User-item historical interactions can be represented as a set $\mathcal{D} = \{(u, i, y_{ui}) | u \in \mathcal{U}, i \in \mathcal{I}\}$, where $y_{ui} = 1$ signifies user u has provided positive feedback on item i, $y_{ui} = 0$ signifies negative

矩阵乘法

对于图中的任意节点对（,）属于V×V，我们可以用一个Np维向量o来表示它们之间的路径关系，其中o的第k个条目表示节点（,）之间第k种类型路径的存在或缺失。

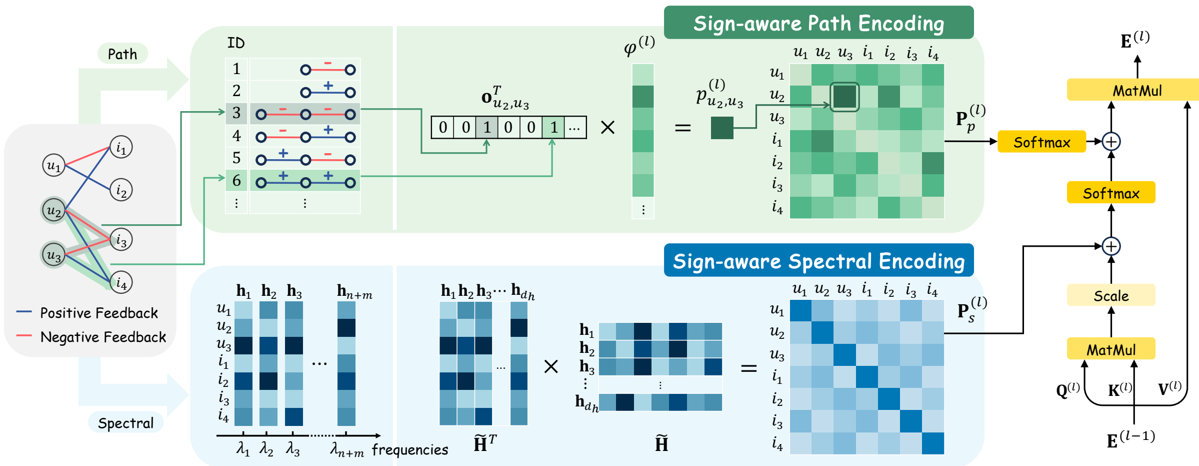

Figure 2: The illustration of our proposed sign-aware path encoding and sign-aware spectral encoding in SIGformer.

feedback, and y_{ui} ='?' signifies an absence of interaction with the item. A signed bipartite graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}^+, \mathcal{E}^-)$ is constructed from $\mathcal{D}$, where the node set $\mathcal{V} = \mathcal{U} \cup \mathcal{I}$ invovles all users and items. The edge sets $\mathcal{E}^+$ and $\mathcal{E}^-$ correspond to user-item positive and negative interactions, respectively, *i.e.*, $\mathcal{E}^+ = \{(u,i)|u \in \mathcal{U}, i \in \mathcal{I}, y_{ui} = 1\}$ and $\mathcal{E}^- = \{(u,i)|u \in \mathcal{U}, i \in \mathcal{I}, y_{ui} = 0\}$. The goal of sign-aware graph-based RS is to learn high-quality embeddings from the signed graph $\mathcal{G}$ and accordingly make accurate recommendation.

For clarity, we introduce some useful notations *w.r.t.* the signed graph. The signed graph can be partitioned into a positive graph $\mathcal{G}^+ = (\mathcal{V}, \mathcal{E}^+)$ and a negative graph $\mathcal{G}^- = (\mathcal{V}, \mathcal{E}^-)$. Let $\mathbf{A}^+$ denote the adjacent matrix of positive graph $\mathcal{G}^+$, where each entry $\mathbf{A}^+_{vw} = 1$ if $(v,w) \in \mathcal{E}^+$ or $(w,v) \in \mathcal{E}^+$; and $\mathbf{L}^+$ denote the Laplacian matrix of $\mathcal{G}^+$, defined as $\mathbf{L}^+ = \mathbf{I} - (\mathbf{D}^+)^{-\frac{1}{2}}\mathbf{A}^+(\mathbf{D}^+)^{-\frac{1}{2}}$, with $\mathbf{D}^+$ representing the diagonal node degree matrix of $\mathcal{G}^+$. Let d_u^+ (or d_i^+) denote the degree of user u (or item i) in the positive graph. Analogous definitions of notations $\mathbf{A}^-, \mathbf{L}^-, \mathbf{D}^-, \mathbf{d}_u^-, \mathbf{d}_i^-$ are applicable to the negative graph $\mathcal{G}^-$.

Compared with the traditional graph-based recommendation that only utilizes $\mathcal{G}^+$, sign-aware recommendation leverages the complete signed graph $\mathcal{G}$, encompassing both positive and negative relations. The signed graph carries richer collaborative information, necessitating effective exploitation by recommendation methods.

2.2 Transformer

特征维度是d，序列长度为n

The transformer architecture has been widely applied in many fields [2, 4, 18, 33, 50, 52]. It is composed of self-attention modules and feed-forward neural networks. In the self-attention module, the input features $\mathbf{X} \in \mathbb{R}^{n \times d}$ are projected to the corresponding query $\mathbf{Q}$, key $\mathbf{K}$, and value $\mathbf{V}$, and then calculated via attention with:

计算注意力分数，过一个softmax，得到注意力矩阵。

$$\mathbf{Q} = \mathbf{X}\mathbf{W}_Q, \quad \mathbf{K} = \mathbf{X}\mathbf{W}_K, \quad \mathbf{V} = \mathbf{X}\mathbf{W}_V,$$
$$\mathrm{Attn}(\mathbf{X}) = \mathrm{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_K}})\mathbf{V} \tag{1}$$

where $\mathbf{W}_Q \in \mathbb{R}^{d \times d_K}, \mathbf{W}_K \in \mathbb{R}^{d \times d_K}, \mathbf{W}_V \in \mathbb{R}^{d \times d_V}$ denote the projected matrices of query, key and value respectively.

Connecting with Collaborative Filtering. The transformer architecture aligns closely with the fundamental principle of collaborative filtering. Specifically, consider the input as features of users and items. The transformer initially estimates the similarity between users and items based on their projected features, then aggregates information from other entities according to this similarity, with more significant contributions from similar entities. This alignment inspires the application of the transformer in sign-aware graph-based recommendation. Nevertheless, the vanilla transformer model cannot be directly adopted, as it fails to harness the structure information of the signed graph. 普通变压器不能利用图的结构信息

Graph Positional Encodings. Positional Encoding has been validated as an effective solution to integrating structural information of the graph into transformer. In recent years, diverse strategies have emerged, including node degrees [65], shortest path distances [38, 65], subgraph representations [6] and spectral features [19, 36]. However, these strategies are not specifically designed for the signed graph, which requires consideration of the sign of edges. Therefore, it is imperative to develop novel positional encodings tailored for the sign-aware recommendation task.

3 METHODOLOGY

In this section, we first provide an overview of SIGformer (Sec 3.1), followed by a description of the proposed positional encodings (Sec 3.2 & Sec 3.3). Finally, we elaborate on implementation details (Sec 3.4).

3.1 Overview of SIGformer

SIGformer employs a transformer architecture for sign-aware recommendation, deviating from the conventional graph-based recommendation paradigm by replacing GNNs with transformer. Specifically, SIGformer comprises the following components:

Embedding Module. As an initial step, each user and item is endowed with a d-dimensional embedding (*i.e.*, $\mathbf{e}_u^{(0)}, \mathbf{e}_i^{(0)}$), which can be treated as learnable parameters or transformed from user/item attributes. For a better description, we collect initial embeddings of all users and items by a matrix:

$$\mathbf{E}^{(0)} = [\underbrace{\mathbf{e}_{u_1}^{(0)}, \cdots, \mathbf{e}_{u_n}^{(0)}}_{\text{user embeddings}}, \underbrace{\mathbf{e}_{i_1}^{(0)}, \cdots, \mathbf{e}_{i_m}^{(0)}}_{\text{item embeddings}}]^T. \tag{2}$$

收集到一个初始的用户项目的embedding矩阵

将图中的结构信息（通过位置编码 P(1)P (1) ）与节点的特征信息结合起来，以便在图神经网络中更好地捕捉节点之间的复杂关系

Sign-aware Transformer Module. Diverging from traditional GNN-based methods, we employ a stack of multi-layer transformer to capture collaborative information. For the l-th layer of the transformer, the embeddings are updated iteratively as follows:

$$\mathbf{Q}^{(l)} = \mathbf{K}^{(l)} = \mathbf{V}^{(l)} = \mathbf{E}^{(l-1)}$$

$$\mathbf{E}^{(l)} = \frac{1}{2}(\text{softmax}(\frac{\mathbf{Q}^{(l)}(\mathbf{K}^{(l)})^T}{\sqrt{d}} + \mathbf{P}_s^{(l)}) + \text{softmax}(\mathbf{P}_p^{(l)}))\mathbf{V}^{(l)} \tag{3}$$

Contrary to the vanilla transformer model, we omit the projected matrices $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V$ as they were found to minimally enhance performance while increasing training difficulty. Besides, we introduce two positional encodings, $\mathbf{P}_s^{(l)}$ and $\mathbf{P}_p^{(l)}$, to explicitly encode the signed graph information, which would be detailed in the next two subsections. Here we separate the two positional encodings into different softmax functions to mitigate the impact of their magnitude disparities.

Prediction Module. Consistent with existing graph-based methods [24, 67], After L layers of transformer, we aggregate the embeddings from each layer to generate the final embeddings:

$$\mathbf{E} = \frac{1}{L+1} \sum_{0 \le l \le L} \mathbf{E}^{(l)} \tag{4}$$

The model prediction is generated from the final embeddings, *e.g.*, through an inner product, a function widely adopted by existing approaches [3, 37, 42, 48]: 将用户和项目的embedding作内积

$$\hat{y}_{ui} = \mathbf{e}_u^T \mathbf{e}_i \tag{5}$$

3.2 Sign-aware Spectral Encoding (SSE)

Graph spectral theory [12, 49] suggests the effectiveness of spectral features (*e.g.*, Laplacian eigenvectors) in capturing the graph structure. Spectral features have also been employed to enhance the GNNs or transformer models on the vanilla graph [36, 47, 55]. Motivated by these successes, we propose to leverage spectral features to enhance our sign-aware transformer model. We begin by combining the Laplacians of the positive and negative graphs as follows: 我们首先结合正负图的拉普拉斯算子，如下所示：

$$\mathbf{L} = \frac{1}{1-\alpha}(\mathbf{L}^+ - \alpha \mathbf{L}^-) \tag{6}$$

where α is a flexible hyperparameter controlling the influence of the negative graph, which will be explored later. The Laplacian eigenvectors of the signed graph are:

$$\mathbf{L} = \mathbf{H}^T \mathbf{\Lambda} \mathbf{H}, \quad \mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_{n+m}]^T \tag{7}$$

最小特征值的特征向量编码图节点关系

where $\mathbf{H}, \mathbf{\Lambda}$ correspond to the eigenvectors and eignvalues respectively. The eigenvectors of the d_h smallest eigenvalues denoted $\tilde{\mathbf{H}}$, are used for encoding node relations in the signed graph:

$$\mathbf{P}_s^{(l)} = \theta^{(l)} \tilde{\mathbf{H}}^T \tilde{\mathbf{H}}, \quad \tilde{\mathbf{H}} = [\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_{d_h}]^T \tag{8}$$

where $\theta^{(l)}$ is a learnable positive parameter for rescaling the magnitude. 重新调整大小：放缩

Connecting with Low-pass Filtering. To elucidate the rationale behind the proposed spectral encoding, we draw a connection to low-pass filtering. For convenience, we omit the softmax function from the analysis as its role is normalization. Similarly, for ease of discussion, we select an arbitrary column of $\mathbf{V}^{(l)}$ for analysis and denote it as $\mathbf{v}$. The vector $\mathbf{v}$ can be expressed by a combination of the basis $\mathbf{H}$:

省略了softmax函数

表示信号强度

$$\mathbf{v} = \sum_{1 \le k \le n+m} \varepsilon_k \mathbf{h}_k \tag{9}$$

where ε_k represents the strength of the signal on the component $\mathbf{h}_k$. The effect of introducing $\mathbf{P}_s^{(l)}$ can be described as:

$$\mathbf{P}_s^{(l)} \mathbf{v} = (\sum_{1 \le k \le d_h} \theta^{(l)} \mathbf{h}_k \mathbf{h}_k{}^T)(\sum_{1 \le k \le n+m} \varepsilon_k \mathbf{h}_k) = \theta^{(l)} \sum_{1 \le k \le d_h} \varepsilon_k \mathbf{h}_k \tag{10}$$

where only the low-frequency components $(h_1, h_2, \cdots, h_{d_k})$ are preserved, and higher-frequency components are filtered out. The lemma below elucidates the efficacy of this low-pass filtering nature:

Lemma 1. *The low-frequency components $(h_1, h_2, \cdots, h_{d_h})$ optimizes the following objective function:*

$$[\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_{d_h}] = \mathop{\arg\min}_{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_{d_h}} \sum_{1 \le k \le d_h} \Big(\underbrace{\sum_{(u,i) \in \mathcal{E}^+} (\frac{\mathbf{z}_{ku}}{\sqrt{d_u^+}} - \frac{\mathbf{z}_{ki}}{\sqrt{d_i^+}})^2}_{\textit{Drawing Positive Neighbors}} \underbrace{- \alpha \sum_{(u,i) \in \mathcal{E}^-} (\frac{\mathbf{z}_{ku}}{\sqrt{d_u^-}} - \frac{\mathbf{z}_{ki}}{\sqrt{d_i^-}})^2}_{\textit{Distancing Negative Neighbors}} \Big)$$

拉进了积极信息，远离了负面信息

$$s.t. \quad \mathbf{z}_k \in \mathbb{R}^{n+m}, \mathbf{z}_k^T \mathbf{z}_k = 1, \mathbf{z}_k^T \mathbf{z}_l = 0, \forall k \ne l, 1 \le k, l \le d_h \tag{11}$$

The proof is presented in Appendix A.1. From the lemma, when $\alpha > 0$, the low-frequency components can be interpreted as the optimal components that minimize the distances between nodes with positive edges while maximizing the distances between nodes with negative edges. Therefore, the introduction of $\mathbf{P}_s^{(l)}$ preserves those desired signals in the embeddings and filters out others, drawing the embeddings of nodes with positive edges closer together and distancing those with negative edges. The structure of the signed graph is thus explicitly encoded into the embeddings.

The Role of α. This lemma also sheds light on the role of the parameter α: it modulates the impact of the negative graph. A larger α implies a stronger emphasis on distancing the neighbors in the negative graph. To enhance the model's flexibility, we propose extending the range of α to include negative values. Interestingly, this simple adjustment proves highly effective. It is based on the intuition that negative feedback may not always be really negative but instead relatively less positive compared to positive feedback [30]. For instance, in a rating system, a user's decision to leave a rating implies engagement with the item, regardless of the rating's polarity. It suggests that the user might prefer the rated item, albeit with a low score, over others they choose not to interact with. Consequently, it may be prudent to set α within the range $(-1, 1)$. Our empirical experiments also validate the optimal of α may be located in small negative values.

尽管分数低，但是也超过了他们选择不去交互的物品

3.3 Sign-aware Path Encoding (SPE)

We further exploit path information within the signed graph, which explicitly reflects the collaborative relations between users and items. Our fundamental intuition is that different path types indicate varying levels of affinity between the nodes they connect. As

不同的路径类型表明它们所连接的节点之间的亲和程度不同。

我们最初根据路径的长度和路径内边的标志枚举所有路径类型，为每个路径类型分配一个唯一的枚举ID。

illustrated in Figure 2, we initially enumerate all path types based on their lengths and the signs of edges within the path, assigning each path type a unique enumerated ID. This ID corresponds to a specific path type. To constrain the potentially vast space of path types, we limit our consideration to paths not exceeding a threshold length L_p, as excessively long paths tend to offer limited collaborative information. Consequently, the total number of path types is $N_p = 2(2^{L_p} - 1)$. Thereby, for any node pair $(v, w) \in \mathcal{V} \times \mathcal{V}$ in the graph, we can represent their path relationships with an N_p-dimensional vector $\mathbf{o}_{vw}$, where the k-th entry of $\mathbf{o}_{vw}$ denotes the presence or absence of the k-th type of path between nodes (v, w). We integrate this rich path information into the transformer architecture to capture nodes' affinity:

路径长度不超过阈值Lp

可学习的参数，它捕获了对应路径所反映的节点亲和力。

$$p_{vw}^{(l)} = \mathbf{o}_{vw}^T \varphi^{(l)} \tag{12}$$

where $\varphi^{(l)} \in \mathbb{R}^{N_p}$ is a learnable parameter capturing node affinities as reflected by the corresponding paths. Differing from existing graph transformer methods that primarily encode the shortest-distance path, our approach considers all path relations, offering a holistic view of node relations. For convenience, we aggregate $p_{vw}^{(l)}$ for all node pairs into a matrix, termed as sign-aware path encoding $\mathbf{P}_p^{(l)}$.

3.4 Implementation Details

采用抽样策略进行采样

Sampling for Acceleration. Given the vast number of user-item combinations in Recommender Systems (RS), traversing all node pairs to calculate attention is computationally prohibitive. To address this challenge, we employ a sampling strategy. Specifically, we utilize a random walk strategy on the signed graph to pick up nodes for aggregation and concurrently record the walked path for computing $\mathbf{P}_p^{(l)}$. For each node $v \in V$, we perform a non-cyclic random walk of length L_p starting from each neighbor of v to sample a set of nodes $\mathcal{S}_v$ associated with the trajectory type. This allows for the rapid updating of user/item embeddings as follows:

V = U ∪I

采样与轨迹类型相关的一组节点S。

$$\mathbf{e}_v^{(l)} = \frac{1}{2} \sum_{w \in \mathcal{S}_v} \Bigg(\mathrm{softmax}\Big(\frac{(\mathbf{e}_v^{(l-1)})^T \mathbf{e}_w^{(l-1)}}{\sqrt{d}} + \theta^{(l)} m_{vw} \Big) + \mathrm{softmax}(\varphi_{t_{vw}}) \Bigg) \mathbf{e}_w^{(l-1)} \tag{13}$$

where m_{vw} is the vw-th entry of the matrix $\mathbf{M} = \tilde{\mathbf{H}}^T \tilde{\mathbf{H}}$, which can be pre-computed; t_{vw} denotes the path type when node w is sampled via the random walker. In this way, the time complexity of our attention module is reduced to $O((n+m)d\hat{N})$, where $\hat{N}$ is the average number of nodes sampled per node. Equipped with this sampling strategy, our SIGformer achieves high efficiency.

Optimization. Referring to recent work [24, 48], BPR loss is adopted for optimizing our SIGformer:

sigmod函数

$$\mathcal{L} = -\sum_{(u,i) \in \mathcal{E}^+} \ln \sigma\left(\hat{y}_{ui} - \hat{y}_{uj}\right) + \sum_{(u,i) \in \mathcal{E}^-} \ln \sigma\left(\beta(\hat{y}_{ui} - \hat{y}_{uj})\right) \tag{14}$$

where for each positive/negative feedback (u, i), we sample an item $j \in \{j \in \mathcal{I} | y_{uj} = `?'\}$ that the user has not interacted with for model optimization; β is a hyperparameter that balances the influence from the negative feedback.

I是项目集合

I是项目矩阵，y（ui）是用户和项目的embedding做内积

Table 1: Statistics of datasets, where "Pos/Neg" denotes the ratio between positive and negative interactions.

Dataset	#Users	#Items	#Interactions	Pos/Neg
Amazon-CDs	51,267	46,464	895,266	1:0.22
Amazon-Music	3,472	2,498	49,875	1:0.25
Epinions	17,894	17,660	413,774	1:0.37
KuaiRec	1,411	3,327	253,983	1:5.95
KuaiRand	16,974	4,373	263,100	1:1.25

4 EXPERIMENTS

In this section, we conduct comprehensive experiments to answer the following research questions:

- **RQ1:** How does SIGformer perform compared with existing methods?
- **RQ2:** What are the impacts of the important components (*e.g.*, , two positional encodings, negative interactions) on SIGformer?
- **RQ3:** How do the hyperparameters affect the model performance?
- **RQ4:** How do different path types capture node similarity?
- **RQ5:** How does the runtime of SIGformer compare with existing methods?

4.1 Experimental Settings

4.1.1 Datasets. We conduct experiments on five real-world datasets, which include both positive and negative feedback: **Amazon-CDs** [43], **Amazon-Music** [43], and **Epinions** [51] are three widely-used datasets containing users' ratings on items from the Amazon and Epinions platforms. We closely refer to recent work [30, 42, 48] and consider the interactions with high ratings (*e.g.,* larger than 3.5) as positive feedback and treat others as negative. **KuaiRec** [22] and **KuaiRand** [23] record user behavior within the Kuai App. For KuaiRec, we focus on the dense dataset for experiments and classify positive and negative feedback based on the ratio of user viewing duration to total video duration. Specifically, ratios equal to or exceeding 4 are considered positive, while those below 0.1 are classified as negative. For KuaiRand, we utilize the pure version and employ "is_click" attribute to classify positive and negative data as suggested by [23]. We adopt a conventional 5-core setting and randomly split the dataset into training set, validation set, and testing set in a ratio of 7:1:2. The dataset statistics are presented in Table 1.

4.1.2 Metrics. Two widely-used metrics *Recall@K* and *NDCG@K* are employed for evaluating the recommendation accuracy. In this work, we simply set $K = 20$ as recent work on graph-based recommendation [3, 24, 67].

4.1.3 Baselines. To comprehensively analyze the performance of SIGformer, we compared it with various graph-based baselines:

1) Unsigned Graph-based Recommendation Methods. The following representative graph-based recommendation methods are included:

- **LightGCN** [24]: the classic graph-based method that leverages linear GNNs for recommendation.
- **LightGCL** [3], **XSimGCL** [67]: the state-of-the-art graph-based methods that enhance LightGCN with contrastive learning.
- **GFormer** [37]: the state-of-the-art method that automates the self-supervision augmentation with transformer architecture.

Table 2: Performance comparison between SIGformer and baselines. The best result is bolded and the runner-up is underlined. The mark '*' suggests the improvement is statistically significant with $p < 0.05$.

		Amazon-CDs		Amazon-Music		Epinions		KuaiRec		KuaiRand	
		Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG
Unsigned Graph-based RS	LightGCN	0.1325	0.0781	0.2725	0.1601	0.0854	0.0510	0.0826	0.0499	0.1197	0.0588
	LightGCL	0.1040	0.0591	0.2921	0.1648	0.0864	0.0516	0.0848	0.0520	0.1291	0.0628
	XSimGCL	0.1346	0.0796	0.2848	0.1683	0.0887	0.0558	0.0863	0.0522	0.1293	0.0641
	GFormer	0.1366	0.0812	0.2807	0.1648	**0.0978**	**0.0602**	0.0864	0.0520	0.1083	0.0532
Sign-aware Graph-based RS	SiReN	0.1369	0.0801	0.2880	0.1725	0.0804	0.0492	0.0826	0.0473	0.1167	0.0571
	SiGRec	0.1092	0.0648	0.1591	0.0896	0.0738	0.0475	0.0497	0.0314	0.1266	0.0699
	PANE-GNN	0.1361	0.0810	0.2691	0.1605	0.0532	0.0301	0.0806	0.0514	0.1066	0.0522
Signed Graph Embedding Methods	SBGNN	0.0183	0.0100	0.0641	0.0325	0.0249	0.0143	0.0797	0.0469	0.0750	0.0361
	SLGNN	0.0283	0.0148	0.1498	0.0788	0.0585	0.0336	0.0865	0.0508	0.1082	0.0520
Graph Transformer	SGFormer	0.0492	0.0275	0.2402	0.1373	0.0588	0.0343	0.0840	0.0504	0.0883	0.0423
	SignGT	0.0231	0.0121	0.1283	0.0666	0.0521	0.0300	0.0861	0.0515	0.0927	0.0439
Our Method	SIGformer	**0.1412***	**0.0828***	**0.3091***	**0.1827***	0.0974	0.0585	**0.0908***	**0.0539***	**0.1494***	**0.0722***
		+3.09%	+1.96%	+5.81%	+5.87%	-0.41%	-2.77%	+5.05%	+3.32%	+15.61%	+3.33%

Considering the similarities between GFormer and SHT [62], and acknowledging that GFormer is a more recent development demonstrating better performance than SHT, we simply take GFormer for comparison.

2) Sign-aware Graph-based Recommendation Methods. The following methods utilize both positive and negative feedback:
- **SiReN** [48]: the classic sign-aware recommendation method that learns two sets of embeddings from positive and negative graphs for recommendation.
- **SiGRec** [30]: the representative work that analyzes the role of the negative graph and accordingly leverages GNNs to learn positive and negative embeddings.
- **PANE-GNN** [37]: the state-of-the-art method that leverages contrastive learning in the sign-aware graph-based recommendation model.

3) Signed Graph Representation Methods. To further validate the effectiveness of our SIGformer, we include the following signed graph representation methods from the field of graph mining. We adapt these methods to recommendation tasks with additional BPR loss.
- **SBGNN** [28]: the highly related work utilizing signed bipartite graph neural networks based on the balance principle [14].
- **SLGNN** [39]: the state-of-the-art signed graph representation method leveraging signed Laplacian graph neural networks.

4) Unsigned Graph Representation with Transformer. Two state-of-the-art unsigned graph transformers are included. Analogously, we adapt these methods to recommendation tasks with additional BPR loss.
- **SGFormer** [60]: the state-of-the-art graph representation method leveraging a simple global attention mechanism.
- **SignGT** [11]: the state-of-the-art graph transformer method that can produce signed attention values in their attention modules.

4.1.4 Parameter Settings. For our SIGformer, we adopt the Adam optimizer and search the hyperparameter with grid search. Specifically, we set the hidden embedding dimension d to 64, which is alignment with recent work [42, 48]. We also draw similar conclusion with their dimensions. We simply set the learning rate to $1e-2$, the weight decay to $1e-4$, the number of eigenvectors d_h to 64, and the layers of transformer to $L = 3$. We search for α in the range of $[-0.8, 0.8]$ with step-size 0.2, and β in the range of $[-1, 1]$ with step-size 0.2. The threshold length L_p is chosen in the range of $\{1, 2, 3, 4, 5, 6\}$.

For the compared methods, we use the source code provided officially and follow the instructions in the original papers to search for the optimal hyperparameters. We have traversed and frequently expanded upon, the entire hyperparameter space suggested by the authors to ensure all compared methods achieve optimal performance.

4.2 Performance Comparison (RQ1)

The performance comparison between our SIGformer and all baselines in terms of $Recall@20$ and $NDCG@20$ is shown in Table 2. Overall, our SIGformer outperforms all compared methods across all datasets with few exceptions. Especially in the dataset KuaiRand, SIGformer achieves impressive improvements — 15.6% and 3.3% in terms of Recall@20 and NDCG@20 respectively. While SIGformer performs slightly worse than GFormer in the dataset Epinions, it still outperforms other baselines. It is worth noting that GFormer exhibits instability and even performs worse than basic LightGCN in KuaiRand.

Comparing with Unsigned RS. Generally speaking, our SIGformer outperforms existing unsigned graph-based recommendation methods, while some baselines have adopted subtle contrastive learning strategies. The reason is that they overlook the negative feedback, which also provides rich collaborative information benefiting recommendation.

Comparing with Sign-aware RS. Our SIGformer consistently surpasses existing sign-aware graph-based recommendation methods, validating the superiority of our sign-aware transformer architecture that fully exploits the entire signed graph, as opposed to methods that separately handle positive and negative graphs. Additionally, GNNs and MLPs might not effectively extract information from the negative graph, potentially causing these sign-aware methods to underperform when compared to unsigned graph-based

Table 3: The results of the ablation study, where positional encodings or negative interactions are removed respectively.

	Negative Interactions?	Spectral Encoding?	Path Encoding?	Amazon-CDs Recall	Amazon-CDs NDCG	Amazon-Music Recall	Amazon-Music NDCG	Epinions Recall	Epinions NDCG	KuaiRec Recall	KuaiRec NDCG	KuaiRand Recall	KuaiRand NDCG
SIGformer-w/o-Neg		✓	✓	0.1349	0.0775	0.2937	0.1738	0.0824	0.0477	0.0708	0.0433	0.1173	0.0545
SIGformer-w/o-En	✓			0.1355	0.0779	0.2932	0.1698	0.0894	0.0526	0.0728	0.0448	0.1413	0.0661
SIGformer-w/o-SPE	✓	✓		0.1380	0.0798	0.2988	0.1744	0.0959	0.0574	0.0862	0.0520	0.1471	0.0697
SIGformer-w/o-SSE	✓		✓	0.1381	0.0812	0.2947	0.1758	0.0945	0.0566	0.0866	0.0515	0.1457	0.0703
SIGformer	✓	✓	✓	**0.1412**	**0.0828**	**0.3091**	**0.1827**	**0.0974**	**0.0585**	**0.0908**	**0.0539**	**0.1494**	**0.0722**

methods. This is evident in the performance of PANE-GNN, SiGRec, and SiReN, which are worse than LightGCN in the Epinions dataset.

Comparing with Signed Graph Embedding Methods. SIGformer consistently surpasses all competing methods in signed graph representation across various datasets. Remarkably, these baselines generally exhibit subpar performance, indicating their inadequacy for recommendation tasks. This outcome can be attributed to two key factors: 1) The majority of existing signed graph methods are predicated on balance theory [14] that triads in a graph should have an even number of negative edges. Such an assumption may be overly rigid given the complex and diverse nature of user preference [48]. 2) These methods often utilize a considerable number of parameters and non-linear modules, which struggle to be effectively trained in RS due to its sparse data.

Comparing with Graph Transformer. Our SIGformer still outperforms these graph transformers. This result validates the effectiveness of our SSE and SPE encodings, which are tailored for sign-aware recommendation. Existing graph transformers neither take the signed relations into consideration nor specifically designed for recommendation.

4.3 Ablation Study (RQ2)

We conduct an ablation study to investigate the effects of different modules in SIGformer. The results in terms of $Recall@20$ and $NDCG@20$ are shown in Table 3, where the two positional encodings and negative interactions are removed respectively.

Effects of positional encodings. As can be seen, when removing spectral (SSE) or path (SPE) positional encodings, we consistently observe the performance drops, *i.e.,* SIGformer-w/o-SSE and SIGformer-w/o-SPE exhibit noticeably inferior performance than SIGformer. This result clearly validates the effectiveness of our positional encodings that capture collaborative information from spectral and path perspectives.

Effect of negative interactions. The removal of negative interactions from the training data results in a noticeable performance decline in our SIGformer model. This outcome reveals the significance of negative feedback, affirming our method's ability to effectively leverage its benefits.

4.4 Role of the parameters (RQ3)

Length Threshold L_p in Sampling. As Figure 3 shows, with L_p increasing, the performance of SIGformer generally exhibits an initial improvement followed by a decline. This is because a larger L_p provides the model with a broader receptive field, which can enhance the model's performance. However, the correlations between higher-order neighbors are weaker than those between lower-order neighbors, so an excessively large receptive field may dilute the impact of lower-order neighbors and may even bring more noise.

Hyperparameter α. This hyperparameter controls the impact of negative interactions in sign-aware spectral encoding. As depicted in Figure 3, SIGformer generally demonstrates an initial enhancement, followed by a decrement as α increases. This trend can be attributed to the role of α in balancing the impacts of positive and negative feedback — an overly large or small influence of the negative aspect is suboptimal. Upon examining the optimal values of α, we observe that it is located at minor negative values for the datasets Amazon-CDs, Amazon-Music, Epinions, and KuaiRand. This can be rationalized by their feedback types, namely rating and click. Although these negatively rated or unclicked items are indeed less preferred by users compared to positive ones, they might still be more favored than items with which users have not interacted. Similar conclusions have been drawn in [30]. However, in the case of KuaiRec, the negative feedback, which signifies users swiftly skipping the item, implies a strong aversion towards these items. These results validate the flexibility of our SIGformer, it can adjust the role of the negative feedback.

4.5 Case Study (RQ4)

To investigate how SIGformer understands different path patterns, we present the top-5 and bottom-5 values of the learned φ from KuaiRec for paths with lengths up to 4, which captures the node similarity indicated by each path type. These results are depicted in Figure 4. Here we simply choose parameters in the first layer for illustration. The majority of these results align with our expectations. For example, path types such as "+" and "$- - +$" reflect high affinity while path types "$- - -+$" and "$-+$" suggest low affinity. Further, given the complexity of user preference, these results also reveal some fresh knowledge, *e.g.,* the path type "$+ - +$" suggests strong positive relations.

4.6 Efficiency Comparison (RQ5)

The running time of SIGformer compared with other baselines on three large datasets are depicted in Figure 5. We can find the transformer equipped with a random-walker-based strategy, does not impose a heavy computational burden. The efficiency of our SIGformer is comparable with SiGRec and LightGCN, and much faster than SiReN, PANE-GNN, and GFormer, which employs heavy MLPs or contrastive learning. As the random walker can be quickly implemented in GPUs, SIGformer sometimes exhibits even higher speed than LightGCN.

5 RELATED WORK

5.1 Graph-based Recommender System

Graph-based methods have drawn significant attention in the field of RS. Compared to traditional collaborative filtering methods such as matrix factorization [35] and autoencoders [41], which only utilize first-order interaction information, graph-based methods can

采样中的长度阈值。如图3所示，随着的增加，SIGformer的性能通常呈现先提高后下降的趋势。这是因为较大的为模型提供了更广泛的接受域，从而可以增强模型的性能。然而，高阶邻居之间的相关性比低阶邻居之间的相关性弱，因此过大的接受场可能会稀释低阶邻居的影响，甚至可能带来更多的噪声。

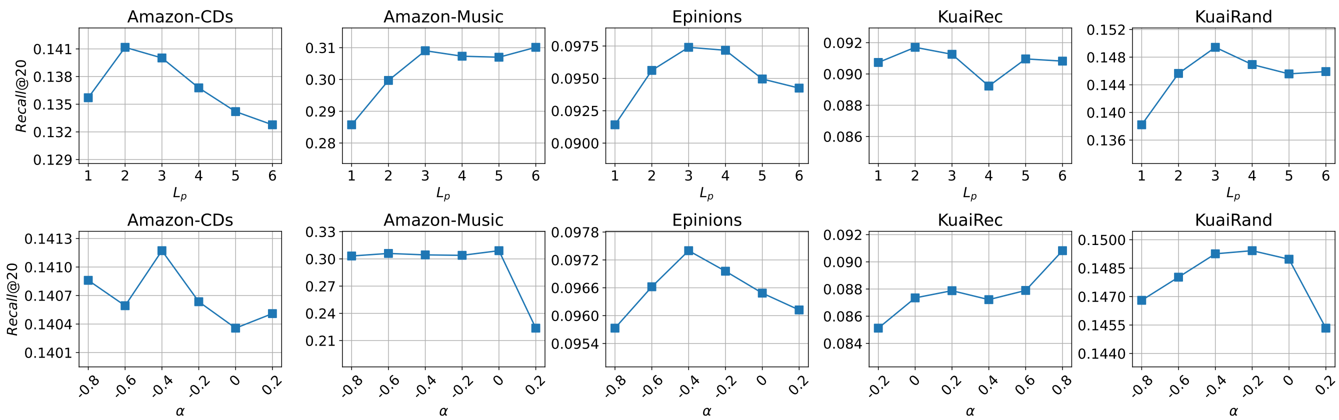

Figure 3: Performance in terms of *Recall*@20 with different K and α.

(a) Top-5 values of φ

Path patterns	Values
+ + − −	1.704
+	1.661
− − +	1.658
+ − +	1.654
+ − −	1.654

(b) Bottom-5 values of φ

Path patterns	Values
− − − +	-1.303
− − + +	-1.226
− +	-1.225
− + − +	-1.210
− + + +	-1.116

Figure 4: The top-5 and bottom-5 values of the learned φ from KuaiRec.

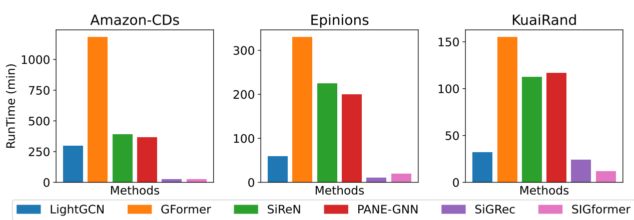

Figure 5: Runtime comparison of SIGformer with baselines.

leverage higher-order relations between nodes in user-item bipartite graph and thus exhibits better performance [61]. In the early years, Wang *et al.*[54], Ying *et al.*[66] and Fan *et al.*[20] directly leveraged graph neural network in recommendation to encode the graph structure information in the representation; LightGCN further [24] simplified the architecture of GCN for recommendation by removing feature transformation and nonlinear activation operations, IMix[13] enhanced the generalizability of GCN by blending interacted and non-interacted item pairs for the same user; subsequently, various graph-based methods emerged that were based on contrastive learning, *e.g.,* SGL [58], LightGCL [3], SimGCL [68], XSimGCL [67], etc; other recent studies identified biases in data used for recommendations[8, 9] and improved existing methods from the perspective of robustness[53, 56, 57]. The transformer architecture has also been utilized by some recent work (*e.g.,* Gformer [37], SHT [62]) to improve the quality of the data augmentation. However, we highlight the following differences between our SIGformer with Gformer and SHT: 1) We directly utilize transformer as the backbone architecture, while the transformer in Gformer and SHT serves as an auxiliary role to generate augmentation; 2) our

主干

SIGformer is tailored for sign-aware recommendation, while they are designed for unsigned graph-based recommendation.

In addition to the aforementioned graph-based methods that predominantly focus on positive data, some studies have explored leveraging both positive and negative feedback in graph-based recommendation. A pioneering example is SiReN [48], which learned both positive and negative embeddings from corresponding graphs through GNNs and MLPs respectively, and subsequently combines these embeddings for recommendation. Huang *et al.*[30] conducted comprehensive analyses to reveal the role of negative feedback and developed a new method SiGRec, which enhances the learning of both positive and negative embeddings via GNNs. PANE-GNN [42] further integrates contrastive learning into negative graph representation learning. Our SIGformer advances beyond these methods in two aspects: 1) Rather than segregating the graph into positive and negative components, we harness the entire signed graph to learn embeddings; 2) We utilize transformer architecture, which is more effective in extracting collaborative information from negative feedback than MLPs and GNNs employed by these previous methods.

5.2 Sign-aware Recommender System

The exploration of sign-aware recommendation traces its origins back to the early years of the field, with initial research predominantly focusing on *explicit feedback*. Explicit feedback, *i.e.,* user ratings, directly indicates users' positive and negative attitudes towards items. This era saw the development of many classic recommendation methods, such as user-based CF [71], PMF [44], and SVD++ [35], etc. However, as the focus of research shifted from explicit to implicit feedback (*e.g.,* clicks, purchases), studies on sign-aware recommendation became less prevalent. Recently, however, due to the availability of negative feedback in many modern RS, sign-aware recommendation has regained significant attention. For instance, the role of negative feedback in RS has been comprehensively investigated in works such as [32, 63]. Negative feedback has also been employed to enhance various recommendation tasks, including graph-based recommendation [30, 42, 48], negative samplers [15, 16], interactive recommendation [70], and sequential recommendation [45, 46], etc.

5.3 Graph Transformer

Transformer [52] has been successfully applied in graph representation learning tasks. Transformer addresses key issues in GNNs, such as over-smoothing [5, 17], over-squashing [1], and limitations in expressive power [64], which stem from the message-passing mechanism that aggregates information from direct neighbors. However, the success of transformer in this domain usually relies on positional encodings that integrate graph structural information into the transformer framework. Recent years have seen a variety of sophisticated designs for positional encodings, including node degrees [65], shortest paths [38, 65], subgraph characteristics [6], edge relations [47], and spectral features [19, 36]. Other works have sought to enhance the transformer model from different perspectives. For instance, SGformer [60] and NAGphormer [10] propose simplified graph transformer models for more efficient and effective representation learning; while SignGT [11] introduces signed attention values to adaptively capture diverse frequency information between node pairs. Despite these advancements, to the best of our knowledge, there remains a notable gap in transformer architectures specifically tailored for the signed graph.

5.4 Signed Graph Representation Learning

Considering both positive and negative edges are available in many applications, signed graph representation learning draws increasing attention. Early strategies on this task include eigen-decomposition of signed Laplacian [26] and matrix factorization [27]. In recent years, research has primarily relied on the balance theory that triads in a graph should have an even number of negative edges [14, 25]. SGCN [14] was the first to extend GCN to the signed graph and designed a new information aggregation and propagation mechanism based on the balance theory for signed networks. SIDE [34] and SIGNET [31] maintained structural balance through random walk strategies. SiGAT [29] and SNEA [40] further designed graph attention mechanisms suitable for signed networks based on the balance theory. SBGNN [28] explored the balance theory in the bipartite graph and proposed a new graph neural network model for learning node representations in the signed bipartite graph. SLGNN[39] returned to spectral graph theory and designed low-pass and high-pass graph convolution filters to extract low-frequency and high-frequency information on positive and negative edges. SBGCL[69] introduced contrastive learning, utilizing dual-level data augmentation to capture explicit and implicit relationships among nodes in signed bipartite graphs, thereby enhancing robustness. However, these methods often can not be directly applied in RS: 1) these methods are often predicated on balance theory, which may be overly rigid given the complex and diverse nature of user preferences [48]; 2) These methods frequently employ a substantial number of parameters and non-linear modules, which struggle to be effectively trained in recommendation systems due to the inherently sparse nature of the data.

6 CONCLUSION AND FUTURE WORK

This study introduces SIGformer, a novel sign-aware recommendation method that utilizes the transformer architecture to comprehensively harness the collaborative information inherent in the signed graph. Within SIGformer, we have innovatively integrated two positional encodings to capture the spectral properties and path patterns of the signed graph. Extensive experiments have been conducted to demonstrate the superior performance of SIGformer over existing graph-based recommendation methods.

A promising direction for future research is the development of a more rapid sign-aware graph transformer architecture. While SIGformer exhibits efficiency, it relies on a sampling strategy that could introduce variance and potentially affect model performance. Additionally, there is significant potential in creating more advanced positional encodings, such as those that fully leverage signed graph spectrum or incorporate additional content information, to further enhance the capabilities of the transformer in RS.

ACKNOWLEDGMENTS

This work is supported by the Starry Night Science Fund of Zhejiang University Shanghai Institute for Advanced Study (SN-ZJU-SIAS-001), OPPO Research Fund, the National Natural Science Foundation of China (62372399), and the advanced computing resources provided by the Supercomputing Center of Hangzhou City University.

A APPENDICES

A.1 The proof of Lemma 1

Proof. According to the properties of symmetric normalized Laplacian,

$$\begin{aligned}
\mathbf{z}_k^T\mathbf{L}^+\mathbf{z}_k &= \mathbf{z}_k^T(\mathbf{I}-(\mathbf{D}^+)^{-\frac{1}{2}}\mathbf{A}^+(\mathbf{D}^+)^{-\frac{1}{2}})\mathbf{z}_k && (15)\\
&= \mathbf{z}_k^T(\mathbf{D}^+)^{-\frac{1}{2}}(\mathbf{D}^+-\mathbf{A}^+)(\mathbf{D}^+)^{-\frac{1}{2}}\mathbf{z}_k && (16)\\
&= \sum_{1\le v\le n+m}\sum_{1\le w\le n+m} A^+_{vw}\Big(\frac{\mathbf{z}_{kv}}{\sqrt{d_v^+}}-\frac{\mathbf{z}_{kw}}{\sqrt{d_w^+}}\Big)^2 && (17)\\
&= \sum_{(u,i)\in\mathcal{E}^+} 2\Big(\frac{\mathbf{z}_{ku}}{\sqrt{d_u^+}}-\frac{\mathbf{z}_{ki}}{\sqrt{d_i^+}}\Big)^2 && (18)
\end{aligned}$$

Similarly, it can be proven that

$$\mathbf{z}_k^T\mathbf{L}^-\mathbf{z}_k = \sum_{(u,i)\in\mathcal{E}^-} 2\Big(\frac{\mathbf{z}_{ku}}{\sqrt{d_u^-}}-\frac{\mathbf{z}_{ki}}{\sqrt{d_i^-}}\Big)^2 \qquad (19)$$

The optimization goal in Eq(11) can be written as:

$$\begin{aligned}
&\arg\min_{\mathbf{z}_1,\mathbf{z}_2,\ldots,\mathbf{z}_{d_h}} \sum_{1\le k\le d_h} \frac{1}{2}(\mathbf{z}_k^T\mathbf{L}^+\mathbf{z}_k-\alpha\mathbf{z}_k^T\mathbf{L}^-\mathbf{z}_k) && (20)\\
=\ &\arg\min_{\mathbf{z}_1,\mathbf{z}_2,\ldots,\mathbf{z}_{d_h}} \sum_{1\le k\le d_h} \frac{1-\alpha}{2}\mathbf{z}_k^T\mathbf{L}\mathbf{z}_k && (21)\\
=\ &\arg\min_{\mathbf{z}_1,\mathbf{z}_2,\ldots,\mathbf{z}_{d_h}} \sum_{1\le k\le d_h} \mathbf{z}_k^T\mathbf{L}\mathbf{z}_k && (22)
\end{aligned}$$

where we have omitted the explicit representation of the constraint for the sake of clarity. The constraint ensures that $\mathbf{z}_1, \mathbf{z}_2, \ldots\mathbf{z}_{d_h}$ form a set of d_h mutually orthogonal unit vectors. Eq(22) represents a constrained quadratic form, which attains its minimum value when $\mathbf{z}_1, \mathbf{z}_2, \ldots\mathbf{z}_{d_h}$ are chosen as the eigenvectors of $\mathbf{L}$ with the d_h smallest eigenvalues.

Therefore, $[\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_{d_h}]$ is a solution to Eq(11). □

公式(22)表示一个受约束的二次型，当选择 z1,z2,⋯,zdaxz 1 ,z 2 ,⋯,z d ax 作为 L 的dh个最小特征值的特征向量时，它达到最小值。

REFERENCES

[1] Uri Alon and Eran Yahav. 2021. On the Bottleneck of Graph Neural Networks and its Practical Implications. In *International Conference on Learning Representations*.

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[3] Xuheng Cai, Chao Huang, Lianghao Xia, and Xubin Ren. 2023. LightGCL: Simple Yet Effective Graph Contrastive Learning for Recommendation. In *The Eleventh International Conference on Learning Representations*.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European conference on computer vision*. Springer, 213–229.

[5] Deli Chen, Yankai Lin, Wei Li, Peng Li, Jie Zhou, and Xu Sun. 2020. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 3438–3445.

[6] Dexiong Chen, Leslie O'Bray, and Karsten Borgwardt. 2022. Structure-aware transformer for graph representation learning. In *International Conference on Machine Learning*. PMLR, 3469–3489.

[7] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. 2021. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 12299–12310.

[8] Jiawei Chen, Hande Dong, Yang Qiu, Xiangnan He, Xin Xin, Liang Chen, Guli Lin, and Keping Yang. 2021. AutoDebias: Learning to debias for recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 21–30.

[9] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2023. Bias and debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems* 41, 3 (2023), 1–39.

[10] Jinsong Chen, Kaiyuan Gao, Gaichao Li, and Kun He. 2022. NAGphormer: A tokenized graph transformer for node classification in large graphs. In *The Eleventh International Conference on Learning Representations*.

[11] Jinsong Chen, Gaichao Li, John E Hopcroft, and Kun He. 2023. Signgt: Signed attention-based graph transformer for graph representation learning. *arXiv preprint arXiv:2310.11025* (2023).

[12] Fan RK Chung. 1997. *Spectral graph theory*. Vol. 92. American Mathematical Soc.

[13] Leyan Deng, Defu Lian, Chenwang Wu, and Enhong Chen. 2022. Graph convolution network based recommender systems: Learning guarantee and item mixture powered strategy. *Advances in Neural Information Processing Systems* 35 (2022), 3900–3912.

[14] Tyler Derr, Yao Ma, and Jiliang Tang. 2018. Signed graph convolutional networks. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 929–934.

[15] Jingtao Ding, Fuli Feng, Xiangnan He, Guanghui Yu, Yong Li, and Depeng Jin. 2018. An improved sampler for bayesian personalized ranking by leveraging view data. In *Companion Proceedings of the The Web Conference 2018*. 13–14.

[16] Jingtao Ding, Yuhan Quan, Xiangnan He, Yong Li, and Depeng Jin. 2019. Reinforced Negative Sampling for Recommendation with Exposure Data.. In *IJCAI*. Macao, 2230–2236.

[17] Hande Dong, Jiawei Chen, Fuli Feng, Xiangnan He, Shuxian Bi, Zhaolin Ding, and Peng Cui. 2021. On the equivalence of decoupled graph convolution network and label propagation. In *Proceedings of the Web Conference 2021*. 3651–3662.

[18] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.

[19] Vijay Prakash Dwivedi and Xavier Bresson. 2021. A Generalization of Transformer Networks to Graphs. *AAAI Workshop on Deep Learning on Graphs: Methods and Applications* (2021).

[20] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. 2019. Graph neural networks for social recommendation. In *The world wide web conference*. 417–426.

[21] Ziwei Fan, Zhiwei Liu, Jiawei Zhang, Yun Xiong, Lei Zheng, and Philip S Yu. 2021. Continuous-time sequential recommendation with temporal graph collaborative transformer. In *Proceedings of the 30th ACM international conference on information & knowledge management*. 433–442.

[22] Chongming Gao, Shijun Li, Wenqiang Lei, Jiawei Chen, Biao Li, Peng Jiang, Xiangnan He, Jiaxin Mao, and Tat-Seng Chua. 2022. KuaiRec: A fully-observed dataset and insights for evaluating recommender systems. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 540–550.

[23] Chongming Gao, Shijun Li, Yuan Zhang, Jiawei Chen, Biao Li, Wenqiang Lei, Peng Jiang, and Xiangnan He. 2022. KuaiRand: An Unbiased Sequential Recommendation Dataset with Randomly Exposed Videos. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 3953–3957.

[24] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 639–648.

[25] Fritz Heider. 1946. Attitudes and cognitive organization. *The Journal of psychology* 21, 1 (1946), 107–112.

[26] Yaoping Hou, Jiongsheng Li, and Yongliang Pan. 2003. On the Laplacian eigenvalues of signed graphs. *Linear and Multilinear Algebra* 51, 1 (2003), 21–30.

[27] Cho-Jui Hsieh, Kai-Yang Chiang, and Inderjit S Dhillon. 2012. Low rank modeling of signed networks. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 507–515.

[28] Junjie Huang, Huawei Shen, Qi Cao, Shuchang Tao, and Xueqi Cheng. 2021. Signed bipartite graph neural networks. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 740–749.

[29] Junjie Huang, Huawei Shen, Liang Hou, and Xueqi Cheng. 2019. Signed graph attention networks. In *Artificial Neural Networks and Machine Learning–ICANN 2019: Workshop and Special Sessions: 28th International Conference on Artificial Neural Networks, Munich, Germany, September 17–19, 2019, Proceedings 28*. Springer, 566–577.

[30] Junjie Huang, Ruobing Xie, Qi Cao, Huawei Shen, Shaoliang Zhang, Feng Xia, and Xueqi Cheng. 2023. Negative can be positive: Signed graph neural networks for recommendation. *Information Processing & Management* 60, 4 (2023), 103403.

[31] Mohammad Raihanul Islam, B Aditya Prakash, and Naren Ramakrishnan. 2018. Signet: Scalable embeddings for signed networks. In *Advances in Knowledge Discovery and Data Mining: 22nd Pacific-Asia Conference, PAKDD 2018, Melbourne, VIC, Australia, June 3-6, 2018, Proceedings, Part II 22*. Springer, 157–169.

[32] Olivier Jeunen. 2019. Revisiting offline evaluation for implicit-feedback recommender systems. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 596–600.

[33] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT*. 4171–4186.

[34] Junghwan Kim, Haekyu Park, Ji-Eun Lee, and U Kang. 2018. Side: representation learning in signed directed networks. In *Proceedings of the 2018 world wide web conference*. 509–518.

[35] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[36] Devin Kreuzer, Dominique Beaini, Will Hamilton, Vincent Létourneau, and Prudencio Tossou. 2021. Rethinking graph transformers with spectral attention. *Advances in Neural Information Processing Systems* 34 (2021), 21618–21629.

[37] Chaoliu Li, Lianghao Xia, Xubin Ren, Yaowen Ye, Yong Xu, and Chao Huang. 2023. Graph Transformer for Recommendation. *arXiv preprint arXiv:2306.02330* (2023).

[38] Pan Li, Yanbang Wang, Hongwei Wang, and Jure Leskovec. 2020. Distance encoding: Design provably more powerful neural networks for graph representation learning. *Advances in Neural Information Processing Systems* 33 (2020), 4465–4478.

[39] Yu Li, Meng Qu, Jian Tang, and Yi Chang. 2023. Signed laplacian graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 4444–4452.

[40] Yu Li, Yuan Tian, Jiawei Zhang, and Yi Chang. 2020. Learning signed network embedding via graph attention. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 4772–4779.

[41] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. 2018. Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 world wide web conference*. 689–698.

[42] Ziyang Liu, Chaokun Wang, Jingcao Xu, Cheng Wu, Kai Zheng, Yang Song, Na Mou, and Kun Gai. 2023. PANE-GNN: Unifying Positive and Negative Edges in Graph Neural Networks for Recommendation. *arXiv preprint arXiv:2306.04095* (2023).

[43] Julian John McAuley and Jure Leskovec. 2013. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *Proceedings of the 22nd international conference on World Wide Web*. 897–908.

[44] Andriy Mnih and Russ R Salakhutdinov. 2007. Probabilistic matrix factorization. *Advances in neural information processing systems* 20 (2007).

[45] Yunzhu Pan, Chen Gao, Jianxin Chang, Yanan Niu, Yang Song, Kun Gai, Depeng Jin, and Yong Li. 2023. Understanding and Modeling Passive-Negative Feedback for Short-video Sequential Recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*. 540–550.

[46] Minju Park and Kyogu Lee. 2022. Exploiting Negative Preference in Content-based Music Recommendation with Contrastive Learning. In *Proceedings of the 16th ACM Conference on Recommender Systems*. 229–236.

[47] Wonpyo Park, Woong-Gi Chang, Donggeon Lee, Juntae Kim, and Seungwon Hwang. 2022. GRPE: Relative Positional Encoding for Graph Transformer. In *ICLR2022 Machine Learning for Drug Discovery*.

[48] Changwon Seo, Kyeong-Joong Jeong, Sungsu Lim, and Won-Yong Shin. 2022. SiReN: Sign-aware recommendation using graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems* (2022).

[49] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. 2013. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine* 30, 3 (2013), 83–98.

[50] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1441–1450.

[51] Jiliang Tang, Huiji Gao, Huan Liu, and Atish Das Sarma. 2012. eTrust: Understanding trust evolution in an online world. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 253–261.

[52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[53] Bohao Wang, Jiawei Chen, Changdong Li, Sheng Zhou, Qihao Shi, Yang Gao, Yan Feng, Chun Chen, and Can Wang. 2024. Distributionally Robust Graph-based Recommendation System. *arXiv preprint arXiv:2402.12994* (2024).

[54] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*. 165–174.

[55] Xiyuan Wang and Muhan Zhang. 2022. How powerful are spectral graph neural networks. In *International Conference on Machine Learning*. PMLR, 23341–23362.

[56] Junkang Wu, Jiawei Chen, Jiancan Wu, Wentao Shi, Xiang Wang, and Xiangnan He. 2024. Understanding contrastive learning via distributionally robust optimization. *Advances in Neural Information Processing Systems* 36 (2024).

[57] Junkang Wu, Jiawei Chen, Jiancan Wu, Wentao Shi, Jizhi Zhang, and Xiang Wang. 2023. BSL: Understanding and Improving Softmax Loss for Recommendation. *arXiv preprint arXiv:2312.12882* (2023).

[58] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised graph learning for recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 726–735.

[59] Liwei Wu, Shuqing Li, Cho-Jui Hsieh, and James Sharpnack. 2020. SSE-PT: Sequential recommendation via personalized transformer. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 328–337.

[60] Qitian Wu, Wentao Zhao, Chenxiao Yang, Hengrui Zhang, Fan Nie, Haitian Jiang, Yatao Bian, and Junchi Yan. 2023. Simplifying and Empowering Transformers for Large-Graph Representations. *arXiv preprint arXiv:2306.10759* (2023).

[61] Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. 2022. Graph neural networks in recommender systems: a survey. *Comput. Surveys* 55, 5 (2022), 1–37.

[62] Lianghao Xia, Chao Huang, and Chuxu Zhang. 2022. Self-supervised hypergraph transformer for recommender systems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2100–2109.

[63] Ruobing Xie, Cheng Ling, Yalong Wang, Rui Wang, Feng Xia, and Leyu Lin. 2021. Deep feedback network for recommendation. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*. 2519–2525.

[64] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2019. How Powerful are Graph Neural Networks?. In *International Conference on Learning Representations*.

[65] Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. 2021. Do transformers really perform badly for graph representation? *Advances in Neural Information Processing Systems* 34 (2021), 28877–28888.

[66] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 974–983.

[67] Junliang Yu, Xin Xia, Tong Chen, Lizhen Cui, Nguyen Quoc Viet Hung, and Hongzhi Yin. 2023. XSimGCL: Towards extremely simple graph contrastive learning for recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2023).

[68] Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Lizhen Cui, and Quoc Viet Hung Nguyen. 2022. Are graph augmentations necessary? simple graph contrastive learning for recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*. 1294–1303.

[69] Zeyu Zhang, Jiamou Liu, Kaiqi Zhao, Song Yang, Xianda Zheng, and Yifei Wang. 2023. Contrastive learning for signed bipartite graphs. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1629–1638.

[70] Xiangyu Zhao, Liang Zhang, Zhuoye Ding, Long Xia, Jiliang Tang, and Dawei Yin. 2018. Recommendations with negative feedback via pairwise deep reinforcement learning. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1040–1048.

[71] Zhi-Dan Zhao and Ming-Sheng Shang. 2010. User-based collaborative-filtering recommendation algorithms on hadoop. In *2010 third international conference on knowledge discovery and data mining*. IEEE, 478–481.