Hindawi

Research Article

Paper Recommendation via Correlation Pattern Mining and Attention Mechanism

Weiming Huang ⓘ**, Baisong Liu** ⓘ**, and Zhaoliang Wang** ⓘ

Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo, Zhejiang 315211, China

Correspondence should be addressed to Baisong Liu; lbs@nbu.edu.cn

In this paper, we improve the efficiency and effectiveness of the matrix factorization method in the paper recommendation system. We mainly address two problems. First, the vectors based on citation networks are undertrained because newly added papers are rarely cited. Second, current algorithms are mainly based on keyword search or global popularity and lack the organic combination of considering personalized interest and global popularity. To address the above two issues for the paper recommender, we propose a matrix factorization model that combines popularity analysis and attention mechanisms. The model effectively fuses the similarity of the citation network and topic using the multiplicative law, which can alleviate the data sparsity problem. Especially for cold-start papers, we add second-order neighbor nodes to makeup for the problem that newly joined papers don't get enough training. We propose a keyword attention mechanism that combines user preferences and global popularity to personalize and balance the popularity of papers. Through comprehensive experiments on the CiteULike dataset, we show that our method can significantly improve the paper recommendation effectiveness.

1. Introduction

Information overload has become a serious problem for scientific researchers. By using the user's profile and metadata from papers, the recommendation system is able to discover and recommend scientific articles that researchers have not yet noticed. Thus, it provides a new way for researchers to find interesting articles beyond simple reference tracking or keyword searching. Various types of recommendation systems have been developed in various digital libraries and academic search engines [1] (e.g., Springer Nature and Elsevier, both of which have their own recommendation systems). Personalized scholarly resource recommendation systems, in particular, are expected to alleviate information overload [1–3] and have emerged as a new research hotspot.

As shown in Figure 1, academic paper recommender system is a context-sensitive recommender system [4]. It is conventional to sort articles by popularity based on user search keywords or keywords from read papers in user profiles. Many various ways have been proposed to improve the effectiveness of the current research paper recommenders. Nascimento et al. [5] offered a method for ranking the

pool of candidate papers after receiving a scholarly publication as input. Because it can only return research papers that closely match the keywords, this strategy is prone to the issues that keyword-based search encounters.

Global popularity [6] can reflect the influence of a paper and gain wide application in recommendation systems as important priori information, but we believe that the combination of global popularity and personalized research interests is still an open problem [7]. For example, a researcher who is interested in recommender systems and is also looking for novel mathematical tools may want the recommender system to support tasks such as "recommending popular papers with mathematical tools that could potentially be applied to the recommender system." Obviously, algorithms based on keyword search or global popularity-based algorithms are not able to meet such personalized popularity requirements.

The most popular recommendation systems are based on collaborative filtering (CF), which is a process of evaluating items by the opinions of other users. In existing CF algorithms, matrix factorization (MF) models play an important role. However, since the distribution of real-world datasets
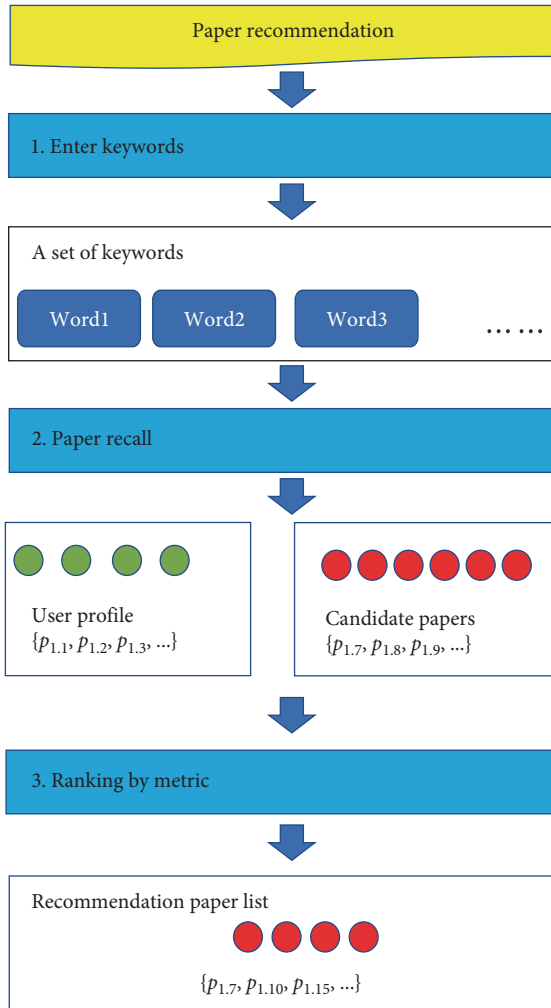
FIGURE 1: A toy example of paper recommendation combining context and user profile. The recommender system recalls articles by keywords entered by scholars, and the system sorts articles by reading user profiles and combining elements such as article content.

obeys a typical power-law distribution, MF models usually suffer from overfitting problems. According to a study by Ganguly and Pudi [8] and Mnih and Salakhutdinov [9], a feasible solution to this problem is to regularize the latent eigenvectors in MF models by means of appropriate priors. There are some efforts to improve rating prediction by introducing citation information through algorithms on graphs. Sugiyama and Kan [10] proposed the use of latent citation augmented data, and Liang et al. [11] defined local and global citation relevance to enhance rating prediction. Other works introduce article topic information, such as in a study by Wang and Blei [12] and Pan and Li [13], regularize the item factor to item topic ratio to improve rating prediction. Jelodar et al. [14] used the trend of topics and the relationship between topics and scholars context documents. Wang et al. [15] proposed a socialized regularized regression model that introduces a priori themes. Pan and Li [13] created a modified item-based recommendation approach based on theme similarity assessment to analyze the contents of research publications. This technique addresses the cold-

start issue, but because it does not consider the reputation of the recommended articles, the suggestions may be less believable, even if their themes are critical.

Collaborative filtering applied to paper recommendations also faces severe data sparsity and cold-start problems. DBLP (website of computer science bibliography, https://dblp.org/) statistics show that the average number of downloads for a paper is 1.6. MF is difficult to make accurate predictions for users with only a few interactions, which is a well-known problem for most existing recommender systems. For MF models, the potential feature vectors of users with only a few interactions are close to the a priori mean [16], and, therefore, the predicted ratings of users with few ratings can be heavily influenced by other users. Even if a citation network is introduced, the same problem of vector undertraining occurs due to the low number of new papers cited [17]. As a result, such a recommendation system will provide inappropriate recommendations, which is typical for scientific article recommendation systems.

We propose a context-aware probability MF model aimed at paper recommendations while alleviating the data sparsity problem and exploiting popularity information. The key idea of our method is to fuse multivariate information by the multiplicative law, which effectively fuses the similarity of citation networks and texts and can alleviate the data sparsity problem. For cold-start papers, we add second-order neighbor nodes to makeup for the problem that newly joined papers don't get enough training. In order to figure out how popular a paper is for a specific person, we suggest a keyword-attention mechanism that takes into account both user preferences and global popularity. We conduct experiments on the CiteULike dataset to evaluate the performance of the proposed model.

Our contributions in this paper are summarized as follows:

(1) We suggest a multiplication rule to combine scores related to the topic, scores related to citations, and scores related to the popularity of keywords into a probability matrix decomposition model to makeup for the lack of data.

(2) We measure the citation relevance of papers by the distance of their embedding vector in paper citation networks. We also introduce second-order neighbors to solve the problem of new papers not being trained enough. We propose a hard attention mechanism to build personal popularity by combining users' keyword preferences and the global popularity of papers.

(3) We conduct extensive experiments on the CiteULike dataset, and the experimental results show that our method is superior to other advanced academic resource recommendation methods.

The remainder of this paper is organized as follows: Section 2 introduces the related work of academic resource recommendation technology, Section 3 proposes our recommendation algorithm, Section 4 introduces the experimental design and performance comparison to verify the effectiveness of the

method, and Section 5 summarizes and discusses the future research work.

2. Related Work

In the current academic big data era, information excess has become a serious problem. Academic search engines rely on keyword search, which is not enough to solve this problem, and personalized recommendation systems for academic resources are expected [2, 3]. System for recommending academic papers is context sensitive. Currently, depending on the means of implementation, current approaches to paper recommendation systems can be classified into three categories: content-based recommendation [18], collaborative filtering-based recommendation [19], and hybrid method-based recommendation [20], and we also provide an overview of popularity used in recommender systems.

2.1. Content-Based Paper Recommendation.
Content-based recommendation uses natural language processing techniques to obtain a representation of a paper text and calculate its similarity to obtain a recommendation list. In recent years, content-based recommendation has become one of the most active research directions in the field of academic paper recommendation systems.

Content-based recommender systems require representation learning and similarity computation for text, which commonly include bag-of-words models, topic models, and word vector techniques. For example, term frequency-inverse document frequency (IDF) [21] is used to map articles into vectors by word frequency-IDF. Topic models such as the commonly used latent Dirichlet allocation (LDA) [22] are used to compute the distribution vectors of article topics by generating models. Deep learning methods such as doc2vec [23] and paper2vec [8] use neural network techniques to generate article representation vectors. Based on the above three methods, the system ranks the papers from highest to lowest by calculating the similarity of the vectors.

The content-based approach is easy to implement because the representation of the text can be computed offline. Such methods, however, lack the variety and novelty of results [3] and ignore the popularity of papers.

2.2. Collaborative Filtering-Based Paper Recommendation.
Collaborative filtering [24] is one of the most successful general recommendation methods. Collaborative filtering makes recommendations by calculating the similarity of interaction records and does not depend on content information. Compared with content-based recommendation methods, collaborative filtering-based recommendations can produce more novel and enlightening recommendation results. Collaborative filtering has achieved great success in traditional recommendation but has encountered difficulties in paper recommendation, where data are extremely sparse. A temporal analysis of indexed data from CiteULike in a study by Bogers and Avd [25] shows that it takes about 2 years for the cold-start problem to disappear and for recommendation performance to improve. But a large number of papers are constantly being added to the system, and these papers are again in cold start.

Due to the aforementioned problems, most of the work on collaborative filtering has been done for sparse data. Wu et al. [24] compared the performance of collaborative filtering and information-rich recommendation. Sugiyama and Kan [10] proposed an adaptive neighbor selection method to alleviate the problem of data sparsity. Sakib et al. [26] used citation relationships of secondary papers to find hidden associations between papers. Liu et al. [27] clustered users based on their standard preferences, and the recommendation output of users was based on the ratings of other users in the close clusters. However, due to the continuous generation of a large number of new papers, it is currently not feasible to use collaborative filtering alone in a paper recommendation system.

2.3. Hybrid Method-Based Paper Recommendation.
Hybrid approaches can combine user behavior records and contextual information to alleviate the cold start problem [28–30], which is the paper recommendation systems' main research direction.

To alleviate the problem of sparse data, introducing multiple a priori information sources is an important idea. Sugiyama and Kan [31] investigated which part of an article can be used to represent a paper. Sun et al. [32] used implicit and explicit social relations to find similar users, including social relations, behavioral relations, and profile similarity. Winoto et al. [33] investigated the introduction of multiple contextual information and the personalized preferences of different types of readers. Wang et al. [34] integrated social relationships and preferences into a standard collaborative filtering model.

VOPRec [35] has some similarities to our paper in that it uses a graph representation learning approach that combines information about the textual content and structural similarity of papers in the citation network. However, VOPRec does not take into account the popularity of the papers, so it cannot clearly show the migration of research hotspots.

2.4. Popularity in Recommendation Systems.
Systems that recommend items try to forecast user preferences so they can show the user things she would like. Different models have been put forth, ranging from nonpersonalized approaches like "Most Popular" (MostPop) [6] to solutions that are tailored to the individual. MostPop, possibly the simplest recommendation algorithm, suggests highly popular items. MostPop is frequently used as a benchmark to offer a reference performance for a recommender system because it is a simple to create and nonpersonalized suggestion approach.

Popularity is a crucial piece of contextual data in text recommendation algorithms. Examples of recommendations that are responsive to popularity include news recommendation [36], book recommendations [37], and tag recommendation [38–40]. Additionally, there are efforts that have started to include popularity as crucial contextual data in paper recommendations. To customize the ranking of papers, literature by Liu et al. [41] combines keyword search with undirected citation graphs. In literature by Ng [42], the results of the title- and abstract-based content similarity metrics, the peer researchers' reviews, the author reputations, and the

TABLE 1: Key symbols in this paper.

Symbol	Symbol definition
U	Users
u_i	A user: $u_i \in U$
V	Papers
v_i	A paper: $v_i \in V$
D	Degree matrix for all papers
d_v	Degree is the number of times an article v is cited: $d_v \in D$
W	Bag of words for articles
w_i	Bag of words for an article: $w_i \in W$
$G_{|U| \times |V|}$	User–article reading record matrix
$G_{|V| \times |V|}$	Adjacency matrix for citation networks
$G_{|V| \times |W|}$	Thesis–keyword matrix

popularity of each candidate article C are all taken into consideration. The popularity of paper C is calculated using the PageRank [43] approach, and the ranking of C is determined using a Borda count voting scheme.

We point out that the MostPop baseline, which is widely used, merely ranks things according to the volume of interactions in the training data. Due to the large number of noninteractive papers, we argue that current popularity assessments (i) are not directly applicable to paper recommender systems and (ii) these methods only concentrate on ranking articles based on popularity under a single keyword, and rarely consider ranking based on multiple keywords.

3. Context-Aware Probabilistic Matrix Factorization

3.1. *Problem Statement and Model Framework.* This section defines the data structure, formulates the research question, and proposes the model framework. Table 1 lists the key notations of this paper.

Definition 1. (Popularity scoring). We define a personalized popularity score that combines popularity and keyword frequency.

Definition 2. (Popularity of unread article). We define a formula that smoothly fit the popularity of uncited papers, as well as heavily cited papers.

Definition 3. (Citation networks). A citation network is a citation relationship between papers, which can be represented by the adjacency matrix of a graph with papers as nodes.

Definition 4. (Neighbors). In a citation network, for a given paper, a paper that is reachable by one hop is its first-order neighbor, a paper that is reachable by two hops is its second-order neighbor, and so on.

As shown in Figure 2, our method aims to generate paper recommendations by combining multiple contextual elements.

First, we calculate the relevance of user-visited and unvisited papers in the citation network using user profiles. The popularity scores of papers are then calculated by combining the keyword hard attention mechanism with the degree of papers in the citation network. Then, using user profiles, we compute topic similarity scores between users and target literature. Finally, we compute the user's interest score for unread papers by combining the above a priori information.

3.2. *Correlation Model of Citation Network Based on Self-Supervised Learning.* In this paper, we consider the citation network as a directed graph network and calculate paper embedding using first-order similarity.

Given the citation network, the recommender system must use as much information as possible to calculate the article's similarity score in the citation network. The model uses an article node's neighbors as contextual information for the current article, assuming that if two articles share more neighbors, their contexts are closer and, thus, more relevant. We create a citation correlation score based on this assumption. To begin, it is assumed that the article, as a node, has its own low-dimensional vector representation p and a contextual low-dimensional vector representation p', with p being close to p' when the article is used as a context.

For any edge $<i, j>$ in the citation network, the conditional probability of generating v_j by node v_i can be expressed, as shown in Equation (1):

$$P\left(v_j'|v_i\right) = \frac{\exp\left(p_k'^{T} * p_i\right)}{\sum_{k=1}^{\|V\|} \exp\left(p_k'^{T} * p_i\right)}, \tag{1}$$

where $\|V\|$ is the number of neighbors.

Two nodes are comparable if their contextual distributions are similar, and the contextual distribution should closely match their empirical distributions. The empirical distribution can be defined, as shown in Equation (2):

$$\widehat{P}\left(v_j|v_i\right) = \frac{1}{d_i}, \tag{2}$$

where the weight of the edge $<i, j>$ is represented by d_i. In this case, the value of d_i is the outdegree of node v_i.

KL divergence is a function that measures the difference between two probability distributions. KL divergence is used as an objective function in the following to measure the difference between the contextual and empirical distributions. The objective function can be simplified, as shown in Equation (3). Negative sampling is used in the model optimization process to reduce computation. The meaning of this formula is that one edge $<i, j>$ in the citation network is sampled at a time as a positive sample, and then k nodes n are sampled from the noise distribution $P_n(v)$ to form k negative samples $<i, n>$. We optimize the objective function using both positive and negative samples. In this paper, we use positive and negative sampling to optimize the objective function.
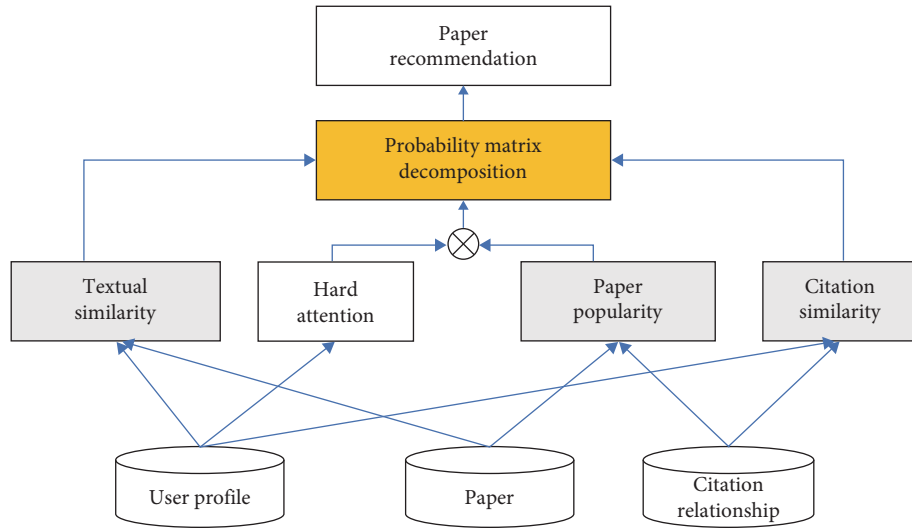
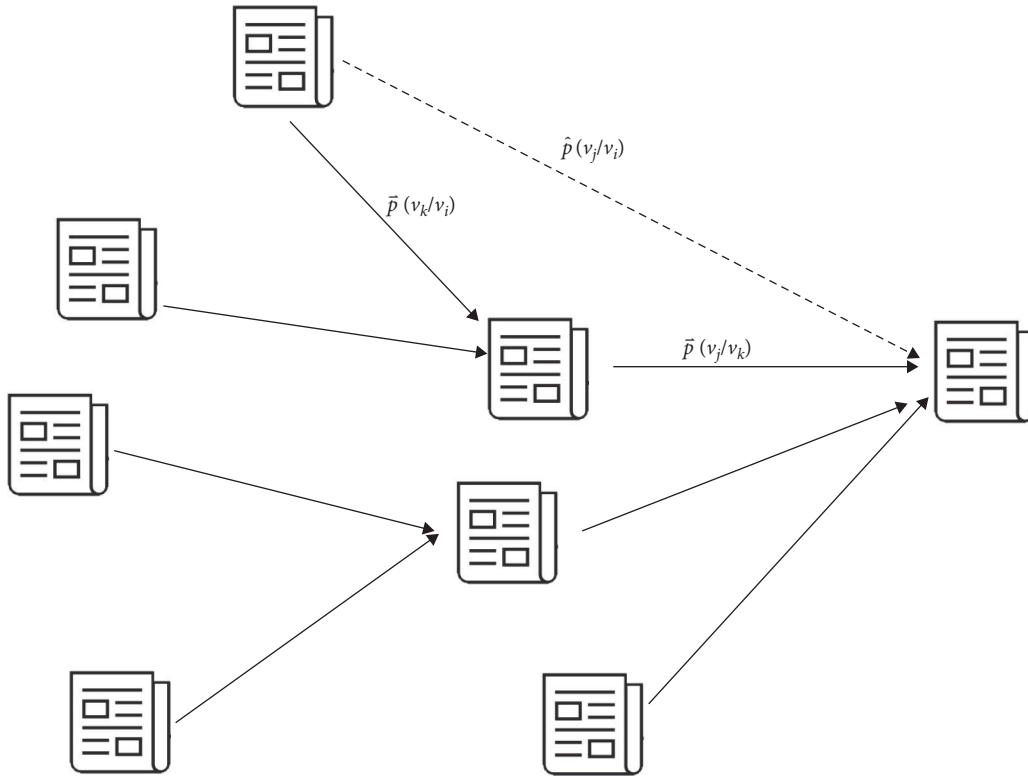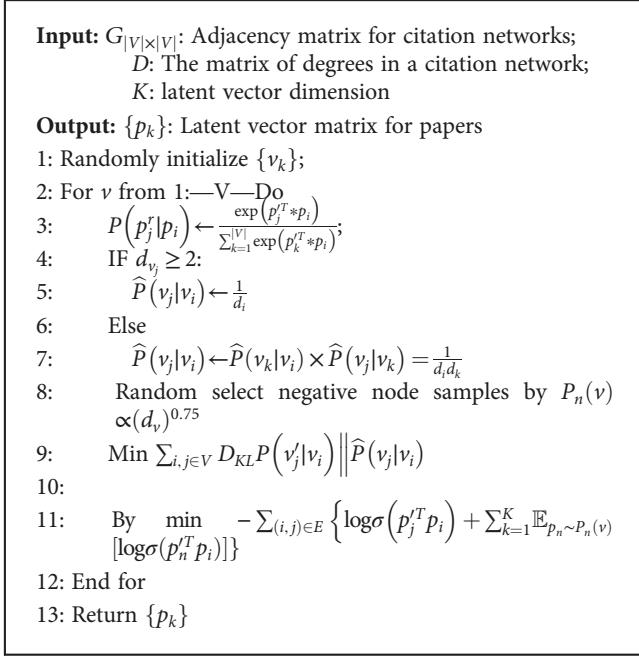FIGURE 2: Context-aware probability matrix factorization.

FIGURE 3: Toy example of citation network embedding.

$$O = - \sum_{(i,j) \in E} \left\{ \log \sigma \left(p_j'^T p_i \right) + \sum_{k=1}^{K} \mathbb{E}_{v_n \sim P_n(v)} \left[\log \sigma (p_n'^T p_i) \right] \right\}. \quad (3)$$

Among them, $\sigma(\cdot)$ is sigmoid function. according to the setting of paper [44], $P_n(v) \propto (d_v)^{0.75}$. $\mathbb{E}$ is the mathematical expectation. Each time, the computational model selects an edge $<i, j>$ from the sampling citation network as a positive sample, and samples K sample nodes from the noise distribution $P_n(v)$ to form a negative sample $<i, n>$.

If there aren't many citations for a new paper, as shown in Figure 3, then the learning of the embedding vector for the new paper isn't good enough. This paper intuitively adopts the sampling of neighbor nodes' neighbors to alleviate this problem. For papers with fewer citations in the citation

Input: $G_{|V| \times |V|}$: Adjacency matrix for citation networks;
       D: The matrix of degrees in a citation network;
       K: latent vector dimension
Output: $\{p_k\}$: Latent vector matrix for papers
1: Randomly initialize $\{v_k\}$;
2: For v from 1:—V—Do
3:      $P\left(p_j^r | p_i\right) \leftarrow \frac{\exp\left(p_j'^T * p_i\right)}{\sum_{k=1}^{|V|} \exp\left(p_k'^T * p_i\right)}$;
4:      IF $d_{v_j} \geq 2$:
5:          $\widehat{P}\left(v_j | v_i\right) \leftarrow \frac{1}{d_i}$
6:      Else
7:          $\widehat{P}\left(v_j | v_i\right) \leftarrow \widehat{P}\left(v_k | v_i\right) \times \widehat{P}\left(v_j | v_k\right) = \frac{1}{d_i d_k}$
8:      Random select negative node samples by $P_n(v)$ $\propto (d_v)^{0.75}$
9:      Min $\sum_{i,j \in V} D_{KL} P\left(v_j' | v_i\right) \left\| \widehat{P}\left(v_j | v_i\right) \right.$
10:
11:     By $\min \; -\sum_{(i,j) \in E} \left\{ \log\sigma\left(p_j'^T p_i\right) + \sum_{k=1}^{K} \mathbb{E}_{p_n \sim P_n(v)} [\log\sigma(p_n'^T p_i)] \right\}$
12: End for
13: Return $\{p_k\}$

ALGORITHM 1: Self-supervised citation embedding.

network, second-order neighbor sampling is used. For example, for the path $i \rightarrow k \rightarrow j$, the empirical distribution is shown in Equation (4):

$$\widehat{P}\left(v_j | v_i\right) = \widehat{P}\left(v_k | v_i\right) \times \widehat{P}\left(v_j | v_k\right) = \frac{1}{d_i d_k}. \tag{4}$$

Our self-supervised citation graph embedding method is summarized in Algorithm 1. Finally, the model calculates the citation network correlation score as $N(u_i, v_j) = \sigma\left(\sum p_j'^T p_i\right)$, where N_{u_i} is the set of papers in user u_i's reading record. In particular, the value of d_{v_j} is read from the adjacency matrix D of the citation network. According to literature by Mikolov et al. [44], we set the power of d_v to 0.75.

3.3. *Methods for Popularity.* The paper's keywords, which unmistakably identify its major concepts, are a significant component. As a result, the article's keywords can be used to categorize it, and the classification's relevance can be used to estimate a user's relevance score for an unselected article. Also, an article's popularity shows the research community's interest in the information it offers. Hence, it is advantageous to take advantage of article suggestions' popularity. We specifically suggest a brand-new attention technique that blends user keyword preferences and article popularity. Popularity analysis and keyword-weighted popularity computation are the two processes in this process.

3.3.1. *Popularity Analysis.* We use the citation network to calculate the objective popularity of an article, and the popularity should not shift in the short term. In this paper, we
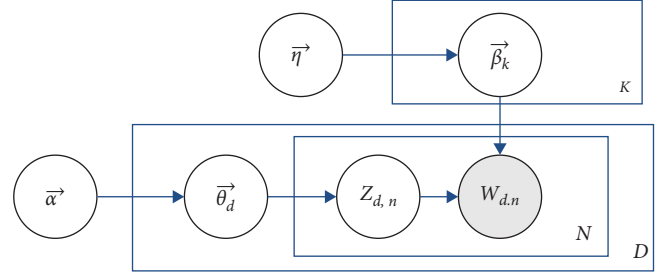

FIGURE 4: Topic model for user profile and papers.

use a weight setting method similar to that in weighted matrix factorization [45] to calculate objective popularity, as shown in Equation (5):

$$p_{v_j} = 1 + \log(1 + 10^e * d), \tag{5}$$

where d is the degree of the article in the citation network. ϵ in the Equation (5) is used to control the rate at which the popularity grows with the number of cited, and a logarithmic function is used so that the popularity does not grow too quickly. This approach ensures that all papers have a non-zero weight on their popularity, and as the number of user visits increases, the popularity of the paper increases accordingly. The popularity of uncited articles is 1.

3.3.2. *Personalize Popularity Scoring Based on Hard Attention Mechanism of Keywords.* We use a hard attention mechanism to personalize the popularity of papers. First, we introduce the hard attention mechanism by defining c_{u_i, v_j} as the keyword preference of user u_i for article v_j, i.e., c_{u_i, v_j} is the ratio of the intersection of user u_i for article v_j keyword c_g to the concurrent set as $c_{u_i, v_j} = \frac{|u_i \cap |v_j|}{|u_i \cup |v_j|}$.

Then, we use the hard attention mechanism to weight the popularity of the unread article v_j, i.e., p_{c_g, v_j}. The keyword popularity p_{u_i, v_j} of user u_i on article v_j is defined, as shown in Equation (6):

$$p_{u_i, v_j} = \frac{c_{u_i, v_j} \cdot p_{v_j}}{\sum_{c_g \in C} c_{u_i, v_j} \cdot p_{v_j}}, \tag{6}$$

where C is the set of keywords in the article collection. A larger value of keyword popularity p_{u_i, v_j} means that more the classification of the article meets the preference of user u_i.

3.4. *Topic Relevance Model*

3.4.1. *Topic Model.* To compute topic similarity, we construct an LDA model, as shown in Figure 4. This model has two latent variables: (1) paper–topic distribution Θ and (2) topic–word distribution B. We believe the topic distribution of papers and their readers is similar. We aggregate the bag-of-words d_{lj} of any paper, and, similarly, we aggregate all the papers read by any user into his user documents and form the bag-of-words d_{ui}, so that we obtain multiple document sets, where each document corresponds to a paper or a user.

The generative process of *LDA* is given as follows:

(1) For each paper and user document, draw topic proportions $\theta_j \sim \text{Dirichlet}(\alpha)$.

(2) For each word n, draw topic assignments $z_{j,n} \sim \text{Mult}(\theta_j)$, draw word $w_{j,n} \sim \text{Mult}(\beta_{z_{jn}})$, and directly calculating θ and β are difficult. Therefore, we estimate their values using variational inference. We define the topic score between user u_i and article v_j as the similarity between user topic distribution θ_i and article topic distribution θ_j. We choose the Jensen–Shannon divergence between user u_i and article v_j, as shown in Equation (7):

$$D_{JS}(u_i, v_j) = \frac{1}{2}D(\theta_i\|M) + \frac{1}{2}D(\pi_i\|M), \quad (7)$$

where $M = \frac{1}{2}(\theta_i + \pi_j)$ and $D(\cdot \| \cdot)$ are the Kullback − Leibler distance. Topic interest-related scores are defined, as shown in Equation (8):

$$S(u_i, v_j) = 1 - D_{JS}(u_i, v_j). \quad (8)$$

The range of $S(u_i, v_j)$ is between 0 and 1, and the closer the article is to the reader's topic interest, the closer the topic score $S(u_i, v_j)$ is to 1.

3.5. Context-Aware Probability Matrix Factorization. We propose a context-aware probabilistic matrix decomposition article recommendation algorithm based on the preceding sections.

3.5.1. Context Fusion. $R = [r_{ui}]$ is the user's reading record matrix for articles. For article recommendation, we use topic information, citation information, popularity information, and keyword information to fit R. We use the multiplication rule [46–48] to fuse topics, citations, and keyword popularity, so as to effectively improved system robustness and cold start performance. Based on the multiplication rule, a unified preference score is defined, as shown in Equation (9):

$$C_{i,j} = S(u_i, v_j) \cdot N(u_i, v_j) \cdot p_{u_i, v_j}, \quad (9)$$

where $S(u_i, v_j)$ is the topic score, $N(u_i, v_j)$ is the citation score, and p_{u_i, v_j} is the personlize popularity score.

3.5.2. Computational Process. Each 1 in $R_{|U|\times|V|}$ means a certain article v_i readed by a user u_i. There are M users

and N articles in total. U_i and V_j are latent feature vectors for users and articles. We define the conditional distribution of observed check-ins, as shown in Equation (10):

$$P(R|U, V, C_{i,j}, \sigma^2) = \prod_{i=1}^{M} \prod_{j=1}^{N} \left[N\left(r_{u_i, v_j} | f\left(U_i, V_j, C_{i,j}, \sigma^2\right)\right) \right]^{I_{ij}}, \quad (10)$$

where $N(\cdot | \mu, \sigma)$ is a Gaussian spherical distribution function with mean μ and variance σ, and I_{ij} is an indicator function ($I_{ij} = 1$ if the user i has read the article j and $I_{ij} = 0$ otherwise). We use the function, as shown in Equation (11), to fit user u_i's check-in to an article v_i.

$$f(U_i, V_j, C_{i,j}) = C_{i,j} \cdot U_i^T V_j. \quad (11)$$

The context is computed in Chapter 3, and we use the weighted inner product of the three to improve the probabilistic matrix factorization (PMF) model.

We set the potential feature vectors of users and articles to a Gaussian spherical prior with zero mean, as shown in Equations (12) and (13).

$$P(U|\sigma_U^2) = \prod_{i=1}^{M} N(U_i|0, \sigma_U^2 I), \quad (12)$$

$$P(V|\sigma_V^2) = \prod_{i=1}^{N} N(V_i|0, \sigma_V^2 I). \quad (13)$$

The posterior distribution is then obtained, as shown in Equation (14), by a straightforward Bayesian inference:

$$
\begin{aligned}
&P(U, V|R, \sigma^2, C_{i,j}, \sigma_U^2, \sigma_V^2) \\
&\propto P(R|U, V, \sigma^2, C_{i,j}, \sigma_U^2, \sigma_V^2) \cdot P(U|\sigma_U^2) \cdot P(V|\sigma_V^2) \\
&= \prod_{i=1}^{M} \prod_{j=1}^{N} \left[N\left(r_{u_i, v_j} | f\left(U_i, V_j, C_{i,j}\right), \sigma^2\right) \right]^{I_{ij}} \\
&\times \prod_{i=1}^{M} N(U_i|0, \sigma_U^2 I) \times \prod_{i=1}^{N} N(V_i|0, \sigma_V^2 I).
\end{aligned} \quad (14)
$$

Consider the posterior distribution of the latent features of users and articles using the logarithmic function, as shown in Equation (15):

$$
\begin{aligned}
&\ln P(U, V|R, \sigma^2, C_{i,j}, \sigma_U^2, \sigma_V^2) \\
&= -\frac{1}{2\sigma^2} \sum_{i=1}^{M} \sum_{j=1}^{N} I_{ij} \left(r_{u_i, v_j} - f\left(U_i, V_j, C_{i,j}\right)\right)^2 \\
&- \frac{1}{2\sigma_U^2} \sum_{i=1}^{M} U_i^T U_i - \frac{1}{2\sigma_V^2} \sum_{j=1}^{N} V_j^T V_j - \frac{1}{2}\left[\left(\sum_{i=1}^{M} \sum_{j=1}^{N} I_{ij}\right) \ln \sigma^2 + MD \ln \sigma_U^2 + ND \ln \sigma_V^2\right] + P,
\end{aligned} \quad (15)
$$

where D represents the dimension of the latent factor and P is a constant that is independent of the parameter. While maximizing the log posterior, the hyperparameters of the latent vectors of users and articles are kept fixed. The optimize procedure is equivalent to minimization of the squared error and a quadratic regularization term of the objective function, as shown in Equation (16):

$$E = \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{N} I_{ij}\left(r_{u_i,v_j} - C_{i,j}\cdot U_i^T V_j\right)^2 \\ + \frac{\lambda_U}{2}\sum_{i=1}^{M}\|U_i\|_F^2 + \frac{\lambda_V}{2}\sum_{j=1}^{N}\|V_j\|_F^2, \quad (16)$$

where $\lambda_U = \frac{\sigma^2}{\sigma_U^2}$, $\lambda_V = \frac{\sigma^2}{\sigma_V^2}$, and $\|\cdot\|_F^2$ are the Frobenius norm. For the optimization of the objective function, as shown in Equation (16), the stochastic gradient descent algorithm is used, as shown in Equations (17) and (18):

$$\frac{\partial E}{\partial U_i} = -\sum_{j=1}^{N} I_{ij}\left(r_{u_i,v_j} - C_{i,j}\cdot U_i^T V_j\right)\cdot \text{context}\cdot V_j + \lambda_U U_i, \quad (17)$$

$$\frac{\partial E}{\partial V_j} = -\sum_{i=1}^{M} I_{ij}\left(r_{u_i,v_j} - C_{i,j}\cdot U_i^T V_j\right)\cdot \text{context}\cdot U_i + \lambda_V V_j. \quad (18)$$

The stochastic gradient descent algorithm is used as follows:

$$U_i \leftarrow U_i - \eta\frac{\partial E}{\partial U_i}, \quad (19)$$

$$V_j \leftarrow V_j - \eta\frac{\partial E}{\partial V_j}. \quad (20)$$

For a given article, our model uses $f(U_i, V_j C_{i,j}) = C_{i,j}\cdot U_i^T V_j$ to predict the user's interest score for it. For cold-start target articles, the calculation is performed based on the context parameters and the global average vector $\bar{U}_l, \bar{V}_j$. Our proposed method is summarized in Algorithm 2.

4. Experiments

4.1. Experimental Setup. We evaluate the performance of our method on academic recommendation dataset.

4.1.1. Dataset Description. CiteULike is a large social networking site that is based on sharing academic literature. Users can upload, recommend, and manage literature on the site. We use the dataset provided by Wang et al. [15]. These datasets are extracted from the open source data on the CiteULike website. The two datasets include user ratings of papers, titles and abstracts of papers, tags of papers, and citation relations of papers, respectively. The citation context of a paper is the most typical contextual relation. As with most datasets in the recommender system domain, the rating

---

Input: $G_{|U|\times|V|}$: User-paper reading record matrix;
$\quad\quad$ $S(u_i, v_j)$: Topic-related score;
$\quad\quad$ p_{u_i,v_j}: Personalized popularity score;
$\quad\quad$ $N(u_i, v_j)$: Citation network-related score
Output: U: User latent vector matrix;
$\quad\quad$ V: Paper latent vector matrix
1: Randomly initialize U, V
2: $C_{i,j} \leftarrow S(u_i, v_j)\cdot N(u_i, v_j)\cdot p_{u_i,v_j}$
3: For each r_{u_i,v_j} in $G_{|U|\times|V|}$ Do
4: $\quad\quad \frac{\partial E}{\partial U_i} \leftarrow -\sum_{j=1}^{N} I_{ij}\left(r_{u_i,v_j} - C_{i,j}\cdot U_i^T V_j\right)\cdot C_{i,j}\cdot V_j + \lambda_U U_i$;
5: $\quad\quad \frac{\partial E}{\partial V_j} \leftarrow -\sum_{i=1}^{M} I_{ij}\left(r_{u_i,v_j} - C_{i,j}\cdot U_i^T V_j\right)\cdot C_{i,j}\cdot U_i + \lambda_V V_j$
6: $\quad\quad U_i \leftarrow U_i - \eta\frac{\partial E}{\partial U_i}$
7: $\quad\quad V_j \leftarrow V_j - \eta\frac{\partial E}{\partial V_j}$
8: End for
9: Return U, V

ALGORITHM 2: Context-aware probability matrix factorization.

TABLE 2: Statistics of datasets.

	CiteULike-a	CiteULike-t
Users	5,551	7,947
Items	16,980	25,975
Tags	46,391	52,946
Citations	44,709	32,565
User-item pairs	204,987	134,860

data in these two datasets are very sparse. We did not make any adjustments to the datasets for the fairness of the comparison. The statistics of the CiteULike dataset are shown in Table 2. As shown in Figure 5, the two datasets conform to the long-tail distribution.

4.1.2. Evaluation Indicators. We use implicit feedback to recommend K articles to users. We use two $Top - K$ metrics to evaluate the quality of the recommended list: Precision @ K and Recall @ K, which are defined as follows.

$$\text{Precision @ K} = \frac{\sum_u |R(u)\cap T(u)|}{K}, \quad (21)$$

$$\text{Recall @ K} = \frac{\sum_u |R(u)\cap T(u)|}{T(u)}, \quad (22)$$

where K is the number of articles recommended to the user, $R(u)$ is the $Top - K$ list of articles recommended to the user, and T(u) is the number of articles actually accessed by the user. In the LDA model, we set $\alpha = 50/T$ and $\beta = 0.1$. For the prevalence score, we found that it is sensitive to the parameter ε and works best when $\varepsilon = 0.5$.

4.1.3. Baselines. The following article recommendation techniques are compared to our method:
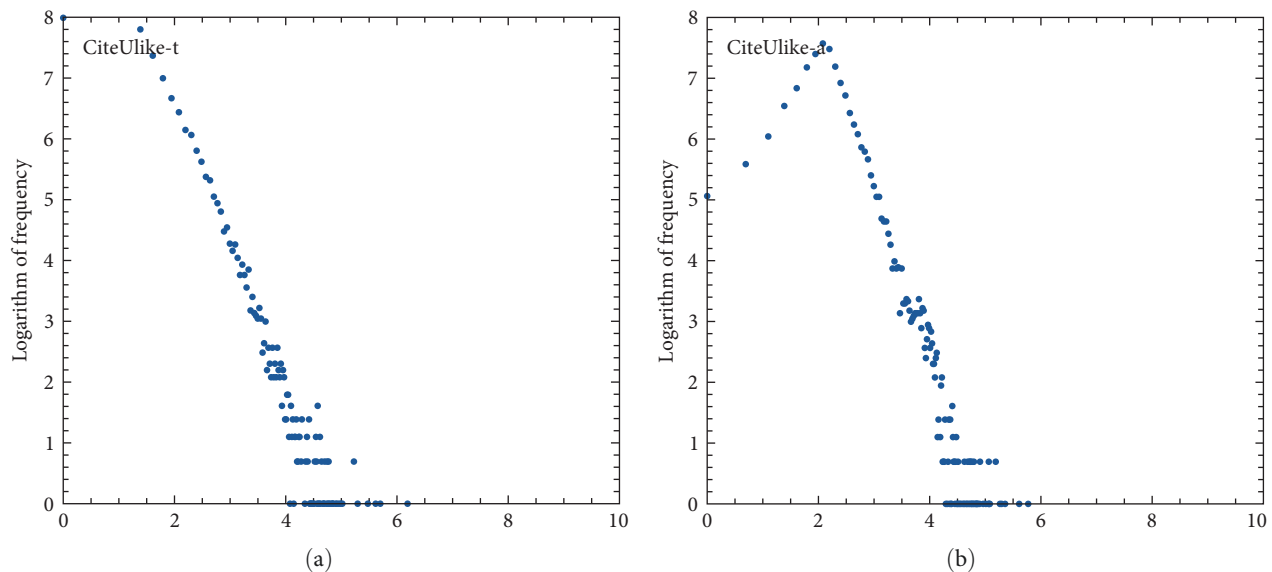
FIGURE 5: Logarithmic frequency distribution of CiteULike dataset. (a) Logarithm of paper reading times. (b) Logarithm of paper reading times.

TABLE 3: Performance comparison.

	CiteULike-a				CiteULike-t			
	Precision@5	Precision@10	Recall@5	Recall@10	Precision@5	Precision@10	Recall@5	Recall@10
MostPop	0.0069	0.0064	0.0168	0.0279	0.0210	0.0139	0.0879	0.1055
UserKNN	0.0227	0.0240	0.0191	0.0429	0.0171	0.0169	0.0287	0.0585
ItemKNN	0.0098	0.0106	0.0067	0.0151	0.0048	0.0059	0.0130	0.0317
PMF	0.0104	0.0094	0.0100	0.0170	0.0157	0.0092	**0.0542**	0.0613
CDR	0.0321	0.0312	0.0262	0.0497	0.0289	0.0256	0.0245	0.0476
CTR	0.0498	0.0455	0.0377	0.0710	0.0378	0.0312	0.0345	0.0695
This study	**0.0514**	**0.0476**	**0.0434**	**0.0807**	**0.0498**	**0.0443**	0.0389	**0.0768**

Bold values signify the optimal result in baselines.

MostPop [49]: MostPop counts the number of occurrences of each item in the training set and ranks the items based on that number.

UserKNN [50]: UserKNN is a recommendation algorithm based on user-based collaborative filtering. The cosine similarity between users is calculated based on their check-in data to the paper.

ItemKNN [51]: ItemKNN is a recommendation algorithm based on item-based collaborative filtering. The cosine similarity is used in itemKNN to calculate the similarity between papers.

PMF [9, 52]: PMF is a traditional MF algorithm that uses a normal distribution prior to represent the hidden eigenvectors with a probabilistic graphical model and Gaussian noise.

CDR [53]: CDR is a hierarchical Bayesian framework for sparsity reduction that combines deep feature representations of item content with user implicit preferences.

CTR [12]: CTR, when combined with MF and topic distribution, can help to solve the problem of data sparsity and article cold start.

We randomly select 80% of CiteULike dataset as training data, and the remaining 20% as test data. For a fair comparison, we refer to the corresponding literature or experimental results of the comparison algorithms to set the parameters of the different algorithms. Under these parameter settings, each comparison algorithm achieves the best performance. In UserKNN and ItemKNN, the number of similar neighbors of users or academic papers is 30; in PMF, $\lambda_U = \lambda_V = 0.001$; we set the learning ratio of PMF $\eta = 0.001$, and the learning ratio of algorithms based on MF academic papers to $\eta = 0.001$.

UserKNN and ItemKNN use the user–paper $0 - 1$ matrix to calculate the similarity between users or papers; PMF, CTR, and our method learn the hidden feature vectors of users and academic papers on the user–paper interaction $0 - 1$ matrix; LDA, CTR, and our method generate topic distributions by learning bags of user words and paper words.

4.2. Experimental Setup

4.2.1. Performance Comparison. Table 3 shows the experimental results of all the comparison algorithms on the two datasets after proper parameter tuning. From Table 3, we can find:

(1) Surprisingly, on both datasets, MostPop, which is based on a simple popularity calculation, shows
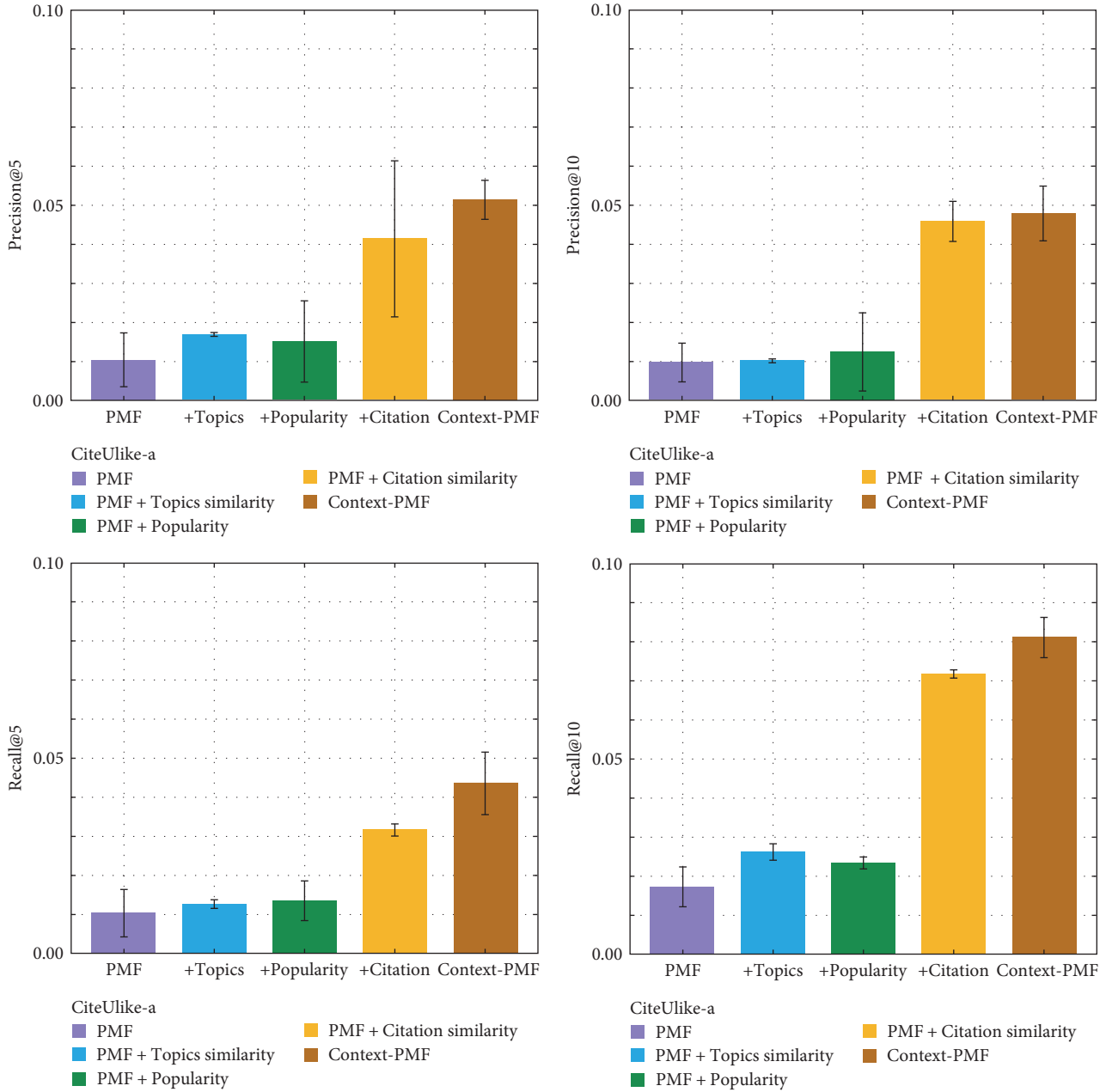
FIGURE 6: Ablation experiment.

higher recall than other collaborative filtering algorithms, suggesting that the popularity of papers in academic paper recommendation algorithms is very important.

(2) In the memory-based algorithm for recommending academic papers, UserKNN has more accurate recommendations than ItemKNN. This is because a large number of academic papers share few user check-in data, so the similarity between academic papers is not as accurate as the calculation of similarity between users. This observation is consistent with the observation in a study by Bogers and Avd [25].

(3) Among all the comparison algorithms, the ItemKNN-based academic paper recommendation algorithm has the lowest accuracy and recall, and the paper-based collaborative filtering cannot fully depict the user's preference for papers.

(4) The model with contextual information makes much better recommendations than the model based on collaborative filtering. This means that the model with context is better for modeling how users interact with academic papers than the model based on collaborative filtering. This observation also shows the potential of models that use more than one piece of
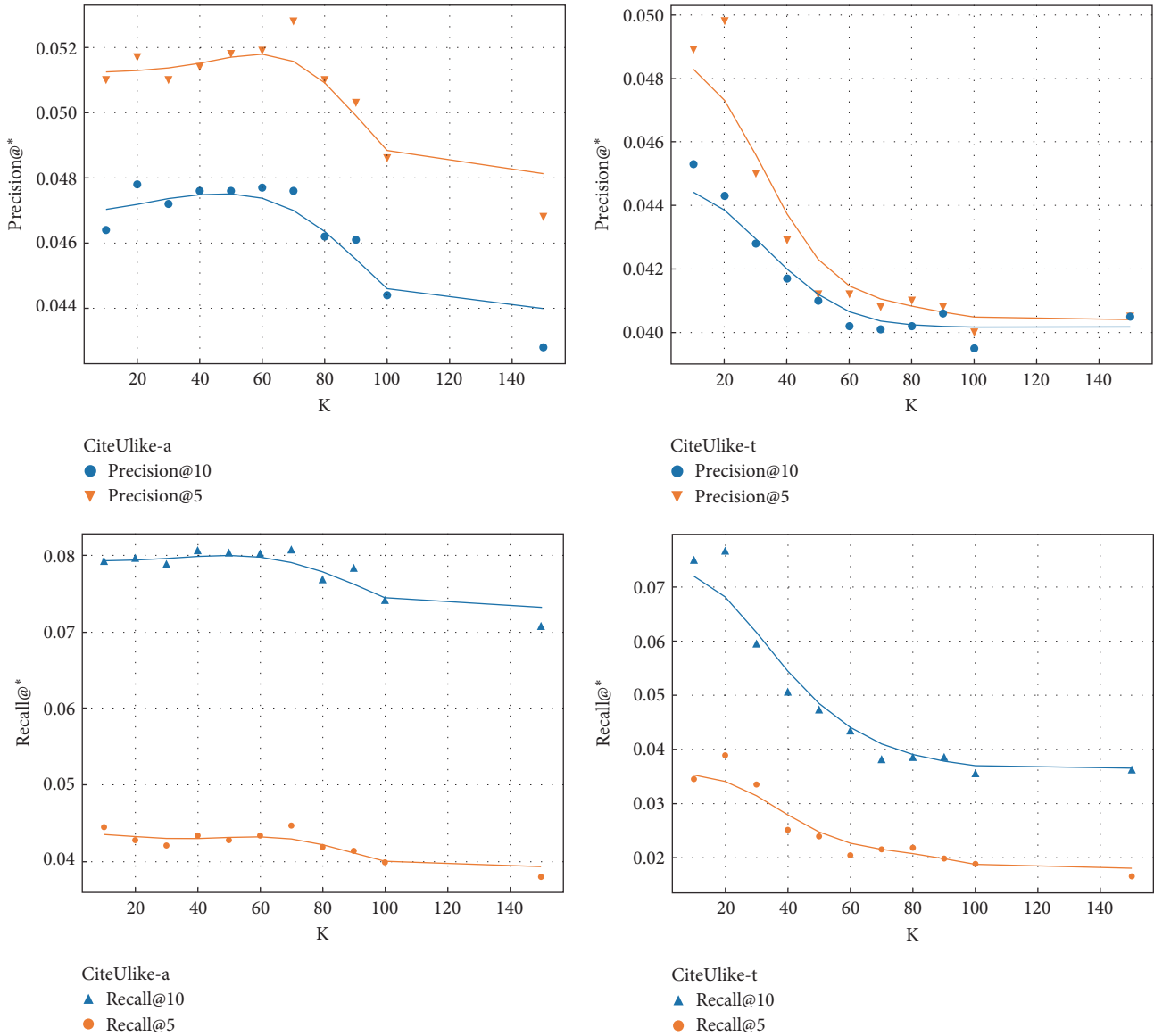
FIGURE 7: Influence of parameter K_d.

context information to help with the task of recommending academic papers.

(5) On the two datasets, the performance of the MF academic paper recommendation algorithm based on context fusion is better than other comparison algorithms, which verifies the effectiveness of our method. Compared with the best results in the comparison algorithms, with Precision@5 as the metric, the improvement rates of the algorithm proposed in this paper on the CiteULike-a and CiteULike-t datasets are 5.1% and 2.2%, respectively; with Recall@5 as the metric, the improvements of the algorithm proposed in this paper on the CiteULike-a and CiteULike-t datasets are 4.2% and 4.7%, respectively. This observation indicates that using contextual information to modify the Gaussian distribution to

model users' check-in behavior can effectively improve the performance of PMF.

(6) On all evaluation metrics, all the comparison algorithms perform better on the CiteULike-a dataset than the CiteULike-t dataset. This is because the average number of users reading academic papers in CiteULike-a is higher than that in CiteULike-t, and the CiteULike-t dataset is more sparse.

4.2.2. Ablation Experiment.
We value the effect of context factors on precision and recall. In this section, we fix the topic score, the popularity score, and the citation score to 1, respectively, and observe changes in precision and recall. Figure 6 illustrates the sensitivity of Precision@5, Precision@10, Recall@5, and Recall@10 to the topic score,

popularity score, and citation score on the CiteULike-a dataset. We can see that the limited introduction of prior information has a positive effect on the indicators, with the correlation between citations having the greatest impact.

4.2.3. Influence of Parameter K_d. The dimensionality K_d of the hidden feature vector is also an important parameter that affects the performance. We change the value of K_d from 10 to 150 in increments of 10 and observe the changes in recall and precision for the CiteULike-a and CiteULike-t datasets. Other parameters are set as: $\lambda_U = \lambda_V = 0.001$, $\eta = 0.001$.

The experimental results are shown in Figure 7. As can be seen from the figures, the values of recall and precision first increase as K_d increases, and after reaching the optimal value, recall and precision decrease as they increase. On the CiteULike-t dataset, the effect of parameter K_d on recall and precision shows a similar trend, but with smaller optimal values. This observation suggests that increasing the value of K_d excessively can cause some problems and introduce some noise that can reduce the accuracy of recommendation algorithm.

5. Conclusion and Future Work

This paper proposes a new context-aware algorithm to address the issues of data sparsity and cold start encountered by the paper recommendation algorithm, as well as the relationship between user interest and paper popularity. The method analyzes the influence of various contextual information on the user's reading behavior and integrates the citation network context, thesis text, keywords, and thesis popularity into the preference acquisition model, so as to extract more accurate user preferences. We also propose a hard attention mechanism to personalize the calculation of the popularity of academic resources, and experimental results on real datasets show that it can significantly improve the recommendation effect.

In recent years, deep learning [54] has shown great advantages in computer vision and natural language processing, and some researchers have combined deep learning techniques with traditional collaborative filtering techniques, such as in the literature by Gündogan and Kaya [20], Wang et al. [55], and Hassan [56], and it would be an interesting research direction to fuse deep learning techniques with our context-aware method.

Data Availability

The data used to support the findings of this study have been deposited at https://github.com/js05212/citeulike-a.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] X. Bai, M. Wang, I. Lee, Z. Yang, X. Kong, and F. Xia, "Scientific paper recommendation: a survey," *IEEE Access*, vol. 7, pp. 9324–9339, 2019.

[2] Z. Li and X. Zou, "A review on personalized academic paper recommendation," *Computer and Information Science*, vol. 12, no. 1, pp. 33–43, 2019.

[3] J. Beel, B. Gipp, S. Langer, and C. Breitinger, "Research-paper recommender systems: a literature survey," *International Journal on Digital Libraries*, vol. 17, no. 4, pp. 305–338, 2016.

[4] K. Sugiyama and M. Y. Kan, "Scholarly paper recommendation via user's recent research interests," in *Proceedings of the 10th Annual Joint Conference on Digital Libraries - JCDL '10*, p. 29, ACM Press, Gold Coast, Queensland, Australia, http://portal.acm.org/citation.cfm?doid=1816123.1816129, 2010.

[5] C. Nascimento, A. H. F. Laender, A. S. da Silva, and M. A. Gonçalves, "A source independent framework for research paper recommendation," in *Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries JCDL '11*, pp. 297–306, Association for Computing Machinery, New York, NY, USA, 2011.

[6] Y. Ji, A. Sun, J. Zhang, and C. Li, "A re-visit of the popularity baseline in recommender systems," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1749–1752, http://arxiv.org/abs/2005.13829, 2020.

[7] C. Sun and Y. Xu, "Topic model-based recommender system for longtailed products against popularity bias," in *2019 IEEE Fourth International Conference on Data Science in Cyberspace (DSC)*, pp. 250–256, IEEE, Hangzhou, China, https://ieeexplore.ieee.org/document/8923895/, 2019.

[8] S. Ganguly and V. Pudi, "Paper2vec: combining graph and text information for scientific paper representation," in *Advances in Information Retrieval Lecture Notes in Computer Science*, J. M. Jose, C. Hauff, I. S. Altıngovde, D. Song, D. Albakour, and S. Watt, Eds., pp. 383–395, Springer, Cham, 2017.

[9] A. Mnih and R. R. Salakhutdinov, "Probabilistic matrix factorization," in *Advances in Neural Information Processing Systems*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds., vol. 20, Curran Associates, Inc., Vancouver, British Columbia, Canadahttps://proceedings.neurips.cc/paper/2007/file/d7322ed717dedf1eb4e6e52a37ea7bcd-Paper.pdf, 2007.

[10] K. Sugiyama and M.-Y. Kan, "A comprehensive evaluation of scholarly paper recommendation using potential citation papers," *International Journal on Digital Libraries*, vol. 16, no. 2, pp. 91–109, 2015.

[11] Y. Liang, Q. Li, and T. Qian, "Finding relevant papers based on citation relations," in *Web-Age Information Management*, H. Wang, S. Li, S. Oyama, X. Hu, and T. Qian, Eds., vol. 6897, pp. 403–414, Springer, Berlin, HeidelbergLecture Notes in Computer Science http://link.springer.com/10.1007/978-3-642-23535-1_35, 2011.p.

[12] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '11*, p. 448, ACM Press, San Diego, California, USA, http://dl.acm.org/citation.cfm?doid=2020408.2020480, 2011.

[13] C. Pan and W. Li, "Research paper recommendation with topic analysis," in *2010 International Conference On Computer Design and Applications*, vol. 4, pp. V4-264–V4-268, 2010.

[14] H. Jelodar, Y. Wang, G. Xiao et al., "Recommendation system based on semantic scholar mining and topic modeling on

conference publications," *Soft Computing*, vol. 25, no. 5, pp. 3675–3696, 2021.

[15] H. Wang, B. Chen, and W. J. Li, "Collaborative topic regression with social regularization for tag recommendation," in *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, F. Rossi, Ed., pp. 2719–2725, IJCAI/AAAI, Beijing, China, August 2013.

[16] Y. Li, M. Yang, and Z. M. Zhang, "Scientific articles recommendation," in *Proceedings of the 22nd ACM international conference on Conference on Information & Knowledge Management - CIKM '13*, pp. 1147–1156, ACM Press, San Francisco, California, USA, 2013.

[17] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "LINE: large-scale information network embedding," in *Proceedings of the 24th International Conference on World Wide Web*, pp. 1067–1077, May 2015.

[18] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The Adaptive Web: Methods and Strategies of Web Personalization*, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds., pp. 325–341, Springer, Berlin, Heidelberg, Berlin Heidelberg, 2007.

[19] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web: Methods and Strategies of Web Personalization*, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds., pp. 291–324, Springer, Berlin, Heidelberg, 2007.

[20] E. Gündogan and M. Kaya, "A novel hybrid paper recommendation system using deep learning," *Scientometrics*, vol. 127, no. 7, pp. 3837–3855, 2022.

[21] D. Dessí, R. Helaoui, V. Kumar, D. R. Recupero, and D. Riboni, *TF-IDF vs Word Embeddings for Morbidity Identification in Clinical Notes: An Initial Study*, SmartPhil@IUI, in, https://api.semanticscholar.org/CorpusID:217164623, 2021.

[22] D. V. Bagul and S. S. Barve, "A novel content-based recommendation approach based on LDA topic modeling for literature recommendation," in *6th International Conference on Inventive Computation Technologies (ICICT)*, pp. 954–961, 2021 https://api.semanticscholar.org/CorpusID:232071650, 2021.

[23] D. Kim, D. Seo, S. Cho, and P. Kang, "Multi-co-training for document classification using various document representations: TF-IDF, LDA, and Doc2Vec," *Information Sciences*, vol. 477, pp. 15–29, 2019.

[24] L. Wu, X. He, X. Wang, K. Zhang, and M. Wang, "A survey on accuracy-oriented neural recommendation: from collaborative filtering to information-rich recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, pp. 4425–4445, 2021, https://api.semanticscholar.org/CorpusID:245502839.

[25] T. Bogers and Bosch Avd, "Recommending scientific articles using citeulike," in *Proceedings of the 2008 ACM Conference on Recommender Systems, RecSys 2008*, P. Pu, D. G. Bridge, B. Mobasher, and F. Ricci, Eds., pp. 287–290, ACM, Lausanne, Switzerland, 2008.

[26] N. Sakib, R. B. Ahmad, and K. Haruna, "A collaborative approach toward scientific paper recommendation using citation context," *IEEE Access*, vol. 8, pp. 51246–51255, 2020, https://ieeexplore.ieee.org/document/9035495/.

[27] L. Liu, N. Mehandjiev, and D. L. Xu, "Multi-criteria service recommendation based on user criteria preferences," in *Proceedings of the fifth ACM conference on Recommender systems - RecSys '11*, p. 77, ACM, Chicago, Illinois, USA, http://dl.acm.org/citation.cfm?doid=2043932.2043950, 2011.

[28] L. Chen and M. Xia, "A context-aware recommendation approach based on feature selection," *Applied Intelligence*, vol. 51, no. 2, pp. 865–875, 2021, https://link.springer.com/10.1007/s10489-020-01835-9.

[29] Y. Wu, Y. ZHao, and S. Wei, "Collaborative filtering recommendation algorithm based on interval-valued fuzzy numbers," *Applied Intelligence*, vol. 50, no. 9, pp. 2663–2675, 2020, http://link.springer.com/10.1007/s10489-020-01661-z.

[30] B. Hui, L. Zhang, X. Zhou, X. Wen, and Y. Nian, "Personalized recommendation system based on knowledge embedding and historical behavior," *Applied Intelligence*, vol. 52, no. 1, pp. 954–966, 2022, https://link.springer.com/10.1007/s10489-021-02363-w.

[31] K. Sugiyama and M. Y. Kan, "Exploiting potential citation papers in scholarly paper recommendation," in *13th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '13*, J. S. Downie, R. H. McDonald, T. W. Cole, R. Sanderson, and F. Shipman, Eds., p. 26, ACM, Indianapolis, IN, USA, 2013.

[32] J. Sun, J. Ma, Z. Liu, and Y. Miao, "Leveraging content and connections for scientific article recommendation in social computing contexts," *The Computer Journal*, vol. 57, no. 9, pp. 1331–1342, 2014.

[33] P. Winoto, T. Y. Tang, and G. I. McCalla, "Contexts in a paper recommendation system with collaborative filtering," *The International Review of Research in Open and Distributed Learning*, vol. 13, no. 5, pp. 56–75, 2012.

[34] G. Wang, X. He, and C. I. Ishuga, "HAR-SI: a novel hybrid article recommendation approach integrating with social information in scientific social network," *Knowledge-Based Systems*, vol. 148, pp. 85–99, 2018, https://www.sciencedirect.com/science/article/pii/S0950705118300820.

[35] X. Kong, M. Mao, W. Wang, J. Liu, and B. Xu, "VOPRec: vector representation learning of papers with text information and structural identity for recommendation," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 1, pp. 226–237, 2021.

[36] N. Jonnalagedda, S. Gauch, K. Labille, and S. Alfarhood, "Incorporating popularity in a personalized news recommender system," *PeerJ Computer Science*, vol. 2, Article ID e63, 2016.

[37] H. Abdollahpouri, M. Mansoury, R. Burke, and B. Mobasher, "The unfairness of popularity bias in recommendation," *ACM*, 2019.

[38] L. Amsaleg, B. Huet, M. Larson et al., "User-aware folk popularity rank: user-popularity-based tag recommendation that can enhance social popularity," in *MM '19: Proceedings of the 27th ACM International Conference on Multimedia*, pp. 1970–1978, Association for Computing Machinery, New York, NY, USA, 2019.

[39] Y. Zhang, J. Hu, S. Sano, T. Yamasaki, and K. Aizawa, "A tag recommendation system for popularity boosting," in *Proceedings of the 25th ACM international conference on Multimedia Mountain View*, pp. 1227-1228, ACM, California, USA, 2017.

[40] Y. Zhang, J. Hu, S. Sano, T. Yamasaki, and K. Aizawa, "Computer vision based and fprank based tag recommendation for social popularity enhancement," in *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*, pp. 1-2, ACM, Tokyo, Japan, 2018.

[41] H. Liu, H. Kou, C. Yan, and Qi L. Keywords-Driven, *Keywords-Driven and Popularity-Aware Paper Recommendation Based on Undirected Paper Citation Graph. Complexity*,

pp. 1–15, https://www.hindawi.com/journals/complexity/2020/2085638/, 2020.

[42] Y. K. Ng, "Research paper recommendation based on content similarity, peer reviews, authority, and popularity," in *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 47–52, IEEE, Baltimore, MD, USA, https://ieeexplore.ieee.org/document/9288359/, 2020.

[43] T. H. Haveliwala, "Topic-sensitive pagerank: a context-sensitive ranking algorithm for web search," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, pp. 784–796, IEEE, 2003.

[44] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in Neural Information Processing Systems*, vol. 26, 2013.

[45] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *2008 Eighth IEEE International Conference on Data Mining*, pp. 263–272, IEEE, Pisa, Italy, http://ieeexplore.ieee.org/document/4781121/, 2008.

[46] C. Cheng, H. Yang, I. King, and M. Lyu, "Fused matrix factorization with geographical and social influence in location-based social networks," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 26, no. 1, pp. 17–23, 2012.

[47] J. D. Zhang and C. Y. Chow, "iGSLR: personalized geo-social location recommendation: a kernel density estimation approach," in *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 334–343, ACM, Orlando, Florida, https://dl.acm.org/doi/10.1145/2525314.2525339, 2013.

[48] B. Liu, Y. Fu, Z. Yao, and H. Xiong, "Learning geographical preferences for point-of-interest recommendation," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1043–1051, ACM, Chicago, IL, USA, https://dl.acm.org/doi/10.1145/2487575.2487673, 2013.

[49] E. Palumbo, G. Rizzo, R. Troncy, E. Baralis, M. Osella, and E. Ferro, "Knowledge graph embeddings with node2vec for item recommendation," in *The Semantic Web: ESWC 2018 Satellite Events - ESWC 2018 Satellite Events, Heraklion, Heraklion, Crete, Greece*, A. Gangemi, A. L. Gentile, and A. G. Nuzzolese, et al., Eds., pp. 117–120, vol. 11155 of *Lecture Notes in Computer Science*, Springer, Cham, Revised Selected Papers 2018, June 2018.

[50] J. Wang, A. P. de Vries, and M. J. T. Reinders, "Unifying user-based and item-based collaborative filtering approaches by similarity fusion," in *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval - SIGIR '06*, p. 501, ACM, Seattle, Washington, USA, http://portal.acm.org/citation.cfm?doid=1148170.1148257, 2006.

[51] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the Tenth International Conference on World Wide Web - WWW '01 Hong Kong*, pp. 285–295, ACM, Hong Kong, http://portal.acm.org/citation.cfm?doid=371920.372071, 2001.

[52] R. Salakhutdinov and A. Mnih, "Bayesian probabilistic matrix factorization using Markov chain Monte Carlo," in *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008)*, W. W. Cohen, A. McCallum, and S. T. Roweis, Eds., pp. 880–887, vol. 307 of *ACM International Conference Proceeding Series*, ACM, Helsinki, Finland, June 2008.

[53] H. Ying, L. Chen, Y. Xiong, and J. Wu, "Collaborative deep ranking: a hybrid pair-wise recommendation algorithm with implicit feedback," in *Advances in Knowledge Discovery and Data Mining - 20th Pacific-Asia Conference, PAKDD 2016*, J. Bailey, L. Khan, T. Washio, G. Dobbie, J. Z. Huang, and R. Wang, Eds., pp. 555–567, vol. 9652 of *Lecture Notes in Computer Science*, Springer, Auckland, New Zealand, April 2016.

[54] Z. Zhang, P. Cui, and W. Zhu, "Deep learning on graphs: a survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 1, pp. 249–270, 2022.

[55] H. Wang, N. Wang, and D. Y. Yeung, "Collaborative deep learning for recommender systems," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining Sydney NSW Australia*, pp. 1235–1244, ACM, New York, NY, USA, 2015.

[56] H. A. M. Hassan, "Personalized Research paper recommendation using deep learning," in *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization Bratislava Slovakia*, pp. 327–330, ACM, New York, NY, USA, 2017.