electronics

MDPI

Article

A Metric Learning Perspective on the Implicit Feedback-Based Recommendation Data Imbalance Problem

Weiming Huang [1,2], Baisong Liu [1,*] and Zhaoliang Wang [1]

[1] Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315211, China; 1801082026@nbu.edu.cn (W.H.); wzl@nbut.edu.cn (Z.W.)

[2] Inner Mongolia Metal Material Research Institute, Baotou 014000, China

[*] Correspondence: lbs@nbu.edu.cn; Tel.: +86-0574-87600233

Abstract: Paper recommendation systems are important for alleviating academic information overload. Such systems provide personalized recommendations based on implicit feedback from users, supplemented by their subject information, citation networks, etc. However, such recommender systems face problems like data sparsity for positive samples and uncertainty for negative samples. In this paper, we address these two issues and improve upon them from the perspective of metric learning. The algorithm is modeled as a push–pull loss function. For the positive sample pull-out operation, we introduce a context factor, which accelerates the convergence of the objective function through the multiplication rule to alleviate the data sparsity problem. For the negative sample push operation, we adopt an unbiased global negative sample method and use an intermediate matrix caching method to greatly reduce the computational complexity. Experimental results on two real datasets show that our method outperforms other baseline methods in terms of recommendation accuracy and computational efficiency. Moreover, our metric learning method that introduces context improves by more than 5% over the element-wise alternating least squares method. We demonstrate the potential of metric learning in addressing the problem of implicit feedback recommender systems with positive and negative sample imbalances.

Keywords: metric learning; context-sensitive; implicit feedback; computational complexity; paper recommendation

check for updates

Citation: Huang, W.; Liu, B.; Wang, Z. A Metric Learning Perspective on the Implicit Feedback-Based Recommendation Data Imbalance Problem. *Electronics* **2024**, *13*, 419. https://doi.org/10.3390/electronics13020419

Academic Editors: Juan M. Corchado, Carlos A. Iglesias, Byung-Gyu Kim, Rashid Mehmood, Fuji Ren and In Lee

Received: 6 December 2023
Revised: 13 January 2024
Accepted: 15 January 2024
Published: 19 January 2024

1. Introduction

Academic recommendation systems have rapidly developed in recent years, and effective academic recommendation systems can alleviate information overload and help researchers quickly find relevant literature. Academic resource platforms have developed content-rich recommendation pages, such as Baidu Scholar, Google Scholar, etc., which provide lists of related paper recommendations.

Content-based, collaborative filtering, and graph-based recommendation systems are the most widely used methods for paper recommendation. Hybrid suggestions utilize two or more different approaches. Content-based approaches compute the text's similarity and produce a recommendation list. Typically, they use techniques like topic modeling [1], word embedding [2], word frequency analysis [3], or a combination of word and sequence modeling approaches [4]. Collaborative filtering-based approaches assess a user's reading records and predict the user's preferences for unread papers using methods such as nearest neighbor computation, matrix decomposition [5], and deep learning [6]. Graph-based approaches, which often use homomorphic graphs, such as citation networks [7–9], or heteromorphic graphs [10,11], like those constructed by entities such as authors–conferences–papers, generate embeddings of the entities. They then create recommendation lists via meta-path methods [12] or graph neural networks [13].

The main focus of this study is implicit feedback-based collaborative filtering. Methods relying on explicit feedback, such as ratings, are not suitable for academic paper recommendations since users do not typically score or rate papers on academic platforms; moreover, suggestions for academic papers should be based on implicit feedback. Compared to explicit feedback, like ratings or reviews, implicit feedback is easier to collect, but it has more uncertainty. Conventional implicit feedback-based recommendation techniques typically depend on subjective negative sample assumptions, such as establishing a negative sample using randomized uniform sampling [14] or based on some a priori information [15]. We contend that the negative sample of academic article recommendations contains outcomes that are ambiguous when based on such assumptions. Since users only have time to read a limited amount of papers, the primary cause of the missing data is a lack of access to the corresponding articles—rather than a lack of interest in the papers' contents. Consequently, the accuracy of recommender systems is restricted by the subjective negative sample assumption. Secondly, there is a significant imbalance in the quantity of positive and negative samples. Because of the system's high quantity of non-interactive papers and the severe data sparsity issue, these papers receive little exposure due to inadequate model training, which hinders the advancement of science and technology as well as the communication of scholarly findings. Some studies introduce context for implicit feedback data in an attempt to reduce sparsity. For example, eALS contends that negative samples should be sorted by hotness [16]. However, we contend that this approach is inappropriate for the academic setting, where we value innovation and require an unbiased method of selecting negative samples.

In this work, we tackle the two aforementioned issues and make improvements from a metric-learning perspective. The following is a summary of this paper's primary contributions:

1. We offer a context-aware metric learning strategy that effectively modifies the model to learn from implicit feedback.

2. The loss function is separately modeled by the algorithm for positive and negative samples. We present the content factor, which improves the data sparsity issue and expedites the objective function's convergence via the multiplication rule for the positive sample pull-in operation. We employ the intermediate matrix caching technique to greatly reduce the computing complexity for the negative sample push-off operation, and we adopt the unbiased global negative sample method.

3. Experimental results on two real datasets show that our method outperforms other baselines in terms of recommendation accuracy and computational efficiency. Our results demonstrate the potential of metric learning in dealing with the problem of implicit feedback recommender systems with positive and negative sample imbalance.

The rest of this paper is organized as follows. Section 2 focuses on the related work, including academic resource recommendations, an implicit feedback-based approach, and metric learning. Section 3 introduces metric learning to model users' implicit feedback and optimizes the computational process by applying the alternating element multiplier method to the negative sampling problems of sparse matrices. Section 4 presents the dataset, experimental methodology, and metrics, introduces the comparison model, analyzes the experimental results, and discusses the contribution of different factors in the model. Finally, Section 5 presents the conclusion of the study and the outlook for future research.

2. Related Work

Personalized academic recommendation systems are an important approach to addressing academic resource overload [17]. This section summarizes the current state of research on the characteristics of recommender systems and academic paper recommendations and reviews some applications of metric learning in recommender systems.

2.1. Classification of Academic Paper Recommendations

Based on the way recommendations are generated, recommendation methods can be categorized into eight groups [18]: collaborative filtering systems, content-based filtering systems, hybrid filtering systems, demographic recommender systems, knowledge-based recommender systems, risk-aware recommender systems, social network recommender systems, and context-aware recommender systems. 'Academic paper recommendation' refers to a subcategory of recommender systems. Current approaches to paper recommender systems can be broadly categorized into content-based, collaborative filtering, and graph-based recommendations. Hybrid recommendations utilize a combination of two or more methods.

Common content-based methods include the word frequency model TF-IDF [3], topic models, such as latent Dirichlet allocation (LDA) [19], and deep learning methods, like doc2vec and paper2vec [20]. Content-based methods can be computed offline, and the results of recommended content are highly relevant, but they tend to offer repetitive recommendations, lacking in diversity and novelty.

Collaborative filtering, which uses the similarity of behavioral records to compute recommendation lists, has been successful in several domains. Collaborative filtering-based recommendation suffers from a problem of under-trained samples, where item vectors with few interaction records are not easily distinguishable in the hidden space. With a large number of new papers in the system, contextual information must be introduced to alleviate this under-training problem [21–23]. Contextual information includes potential citations [24], social connections [25,26], personalized preferences [27], and so on. Reference [28] employs a textual and structural fusion feature as an example.

Typically, graph-based paper recommendations belong to social network recommender systems. The graph-based approach relies on the similarity of nodes on the graph. Much work is based on citation networks, which are homogeneous networks [29], containing only one type of entity and relation. In addition, some works investigate heterogeneous academic networks by analyzing distinct entities, such as authors, venues, and publications [30].

Calculating the similarity between two given papers is essential for predicting citation connections. Recently, methods for learning network representations, which encode structural information about a graph for citation recommendations, have been developed [28]. Zhu et al. [31] proposed a heterogeneous knowledge embedding-based attentive RNN for recommending scientific papers and citations, based on the user's identity and a constrained query length. Li et al. [30] examined meta-paths in the network to determine user preferences and they employed random walks on these meta-paths to calculate the recommendation scores of candidate papers for target users. Using Bayesian personal ranking (BPR) [15] as the objective function, they employed a personalized weight-learning procedure to determine a user's personalized weights on different meta-paths. However, the negative sample noise problem in the paper recommendation system has gone unnoticed. In addition, the graph-based approach to new entry articles suffers from a long-ignored undertraining issue.

2.2. Recommendations Based on Implicit Feedback

Handling missing data is crucial for learning from implicit data in recommendation systems, as they provide a valuable negative signal. Matrix factorization (MF) is a method for representing a data matrix as two low-dimensional matrices. The decomposition procedure can extract data co-occurrence patterns [32]. In addition, the reconstructed low-rank model can be used to recover missing data, such as its applicability in predicting users' ratings on unknown items [33]. Based on how negative samples are handled, previous research can be divided into two categories: sample-based learning and whole-data-based learning.

The first type uses a homogenized random sampling method or a rule-based sampling method to extract negative instances from absent data. For instance, the Bayesian personalized ranking (BPR) method proposed by Rendle et al. [15] randomly samples negative instances from missing Bayesian personalized ranking entries to maximize the margin between the model prediction of observed entries and that of sampled negatives. Recently, He et al. [34] developed adversarial training methods for BPR to enhance the model's robustness. Negative sampling significantly reduces the number of negative instances, thereby controlling the overall time complexity. However, while their convergence process is faster, their performance is highly dependent on the design of the sampler.

The second type considers all absent entries to be negative instances. For instance, Hu et al. [35] modeled all missing entries as negative instances with the label 0 and assigned them a decreased weight. Recent research by Ding et al. [36] established a pairwise learning framework to model the difference between observed entries and all missing entries. These methods model negative instances with greater coverage, but the learning algorithm may be very slow.

Introducing context to complement machine learning algorithms is a classic but never outdated practice [37,38]. He et al. [14] believe that when all other factors are equal, popular items are more likely to be known by users in general; thus, it is reasonable to assume that a miss on a popular item is more likely to be truly irrelevant (as opposed to unknown) to the user. He et al. proposed the use of a popularity-aware weighting strategy for negative samples and used the idea of caching matrices to reduce the amount of computation. In further work, He et al. [16] proposed dynamic negative sample weighting methods for more themes. Their work addressed a research gap by devising efficient learning algorithms for any weighting scheme on missing data. However, most of the above methods are based on collaborative filtering and do not fully utilize the rich contextual information provided by academic platforms, which is the second problem.

In pursuit of model effectiveness and thorough mining of contextual data, we emphasize whole-data-based learning in this work with the aim of developing an effective solution to address inefficiencies. Unlike the aforementioned work, we introduce the contextual information factor from the perspective of metric learning, which similarly improves the model's performance and computational efficiency.

2.3. Metric Learning and Recommender Systems

Recently, recommendation algorithms based on metric learning have attracted considerable attention. Metric learning is able to find an appropriate distance metric between users and items. Based on the Euclidean distance metric, Hsieh et al. [39] first proposed collaborative metric learning (CML), which is similar to Bayesian sorting; it adjusts the distances between users and positive samples, as well as users and negative samples, through drawing-in and pushing-out operations. It models user–item similarity while also considering user–user and item–item similarities. The model was successful, and subsequent studies used it as a benchmark for corresponding improvements. For example, Tay et al. [40] improved the potential user–item interactions using memory networks and attention mechanisms to improve the convergence problem of the model.

Optimization strategies for metric learning are mostly based on loss functions with uniform sampling. Compared to the field of computer vision [41,42], where difficult sample mining is widely used to improve the positive and negative sample imbalance in metric learning, difficult samples in recommender systems are difficult to survey. Tran et al. [43] proposed the use of the two-stage negative sampling method, instead of uniform sampling, to mine informative-containing triples by secondary filtering. Zhang et al. [44] adjusted the distance of negative samples with an adaptive distance function.

The large means nearest neighbor (LMNN) [45] is a widely adopted pairwise loss function that incorporates a ternary loss, including a "push" operation and a "pull" operation. This idea is widely adopted by subsequent models, such as CML, with a loss function as in Equation (1):

$$L_m(d) = \sum_{(i,j)\in R} \sum_{(i,k)\notin R} \left[\alpha + d(i,j)^2 - d(i,k)^2\right] \tag{1}$$

where $d(i,j)$ is the distance between the user, u_i, and positive sample v_j, and $d(i,k)$ is the distance between the user, u_i, and positive sample v_k. R is the record of interaction between the user and the item. The ternary loss is an intuitive loss function that "pushes" negative pairs of samples away from positive pairs of samples by a distance greater than or equal to the interval α.

Although ternary loss has seen good success in metric learning, the model still suffers from the following problems: First, the method of negative sampling is not specified in the loss function. To avoid excessive computational overhead, traditional metric learning models adopt a random sampling method for negative samples, and the ternary information content collected in this way is random. In the severely unbalanced essay recommendation dataset, pure random sampling leads to the selection of triples, which are samples with little informative content, which minimally benefits the model. Second, for essay recommendation datasets, a large amount of content information is non-negligible a priori information. Methods relying solely on user behavior data obviously underutilize the data. In this paper, we conduct research on metric learning to address these two issues.

3. Context-Aware Element-Wise Alternating Least Squares

3.1. Problem Statement and Modeling Framework

This section defines the data structure, describes the research problem, and shows the modeling framework. Table 1 lists the key notations of this paper.

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

In the following, we build the model from a metric learning perspective, discuss the optimization of the model, and finally describe a method for computing the context factor.

As shown in Figure 1, our approach aims to combine multiple contexts to generate paper recommendations. First, we use user profiles to compute the relevance of the user's visited and unvisited papers in the citation network. Then, the user profile is utilized to compute the topic similarity score between the user and the target paper. Finally, we combine the above a priori information to compute the context factor of the user's unread papers.

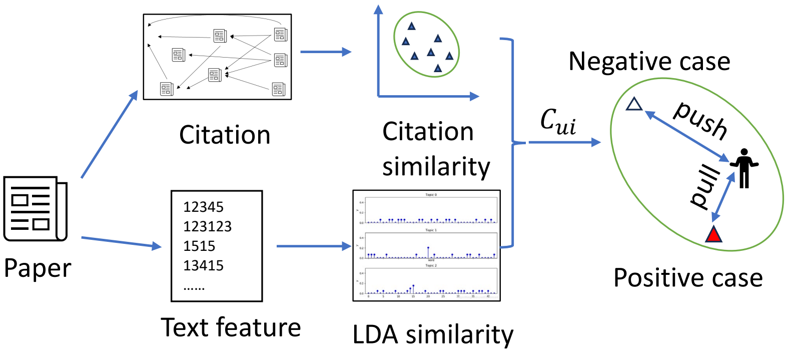

Figure 1. Context-aware paper recommendation by metric learning based on push and pull operations. Context factors were used to speed up the pull operation for positive samples to accelerate convergence. For negative samples, unbiased global sampling was utilized.

3.2. Starting with Metric Learning

Referring to the simple idea of metric learning, we build the objective function based on the “push” and “pull” operations:

$$min_{u^*,v^*}\ L_{push}(s) + \omega\ L_{pull}(s) \tag{2}$$

$$s.t.\ ||p_u^*|| \leq 1\ and\ ||q_i^*|| \leq 1 \tag{3}$$

3.3. User Preference Modeling and Algorithm Optimization

$L_{pull}(s)$ as the “pull operation”, we take the dot product as the distance measure; the closer the dot product is to 1, the closer the vectors are, subject to the modulus constraints$||p_u^*|| \leq 1\ and||q_i^*|| \leq 1$. We take $L_{pull}(s) = \sum_{(i,j\in R)} ||1 - {p_u}^T q_i||_2^2$. For positive feedback from users, the user is clearly influenced by contextual information. For papers that the user has not accessed, it is predominantly because the user is unaware of the paper’s existence. Of course, there are users who have already seen the paper but are not interested. In this case of uncertainty about the user’s interest preference, it is not appropriate to use contextual information to determine the user’s reading. Therefore, in this paper, we refer to eALS [3] and TGSC-PMF [31], and use different fusion strategies to adopt varied contextual information strategies for the two cases of already-read and unread papers. To accelerate convergence, a content relevance factor C_{ui} is introduced to perform a stretching operation on the dot product.

$$L_{pull}(s) = \sum_{(u,i\in R)} ||1 - C_{ui}{p_u}^T q_i||_2^2 \tag{4}$$

where $||C_{ui}||_2^2 \leq 1$; the more relevant the content is, the closer $||C_{ui}||_2^2$ is to 1. When $||C_{ui}||_2^2$ is much smaller than 1, the model enables p_u and q_i to converge to each other’s neighborhood at a faster rate. This introduction of contextual parameters is intuitive: the fact that user u chooses item i with little content relevance suggests that, in some way, this item particularly fits the user’s needs. By quickly bringing such positive sample pairs closer together, the system can quickly capture the user’s particular preferences.

Similarly, in this case of uncertainty about the user’s interest preference, it is inappropriate to use contextual information to determine the user’s reading interest. The closer the dot product is to 0, the further away the vectors are. Therefore, without introducing the content relevance factor in the “push” operation, then $L_{push}(s) = \sum_{(u,i\notin R)} ||0 - {p_u}^T q_i||_2^2$.

ω_{ui} uses the weight setting method of ALS [32], $\omega_{ui} = 1 +\ \log\left(1 + 10^{\varepsilon} r_{ui}\right))$. Then, the objective function can be abbreviated as

$$L = \sum_{(u,i\in R)} \omega_{ui}||1 - C_{ui}{p_u}^T q_i||_2^2 + \sum_{(u,i\notin R)} ||0 - {p_u}^T q_i||_2^2 \tag{5}$$

$$s.t.\ ||p_u^*|| \leq 1\ and\ ||q_i^*|| \leq 1 \tag{6}$$

According to the Lagrange multiplier method, the objective function can be optimized in the following way:

$$L = \sum_{(u,i\in R)} \omega_{ui}||1 - C_{ui}{p_u}^T q_i||_2^2 + \sum_{(u,i\notin R)} ||0 - {p_u}^T q_i||_2^2 + \lambda||p_u||_2^2 +\ \lambda||q_i||_2^2 \tag{7}$$

We can optimize the objective function using the stochastic gradient descent (SGD) method. We set $\hat{r}_{ui} = p_u^T q_i, \widehat{r_{ui}^f} = \hat{r}_{ui} - p_{uf}q_{if}, \hat{r}_{ui} = \widehat{r_{ui}^f} + p_{uf}q_{if}$. Using the derivative of the objective function L with respect to u_{if}, the following equation is obtained: $\frac{\partial L}{\partial p_{uf}} = 2\sum_{(u,i)\in R} \omega \left(1 - C_{ui}\widehat{r_{ui}^f} + C_{ui}p_{uf}q_{if}\right) * \left(-\alpha q_{if}\right) + 2\sum_{(u,i)\notin R} \left(\widehat{r_{ui}^f} + p_{uf}q_{if}\right) q_{if} + 2\lambda p_{uf}$.

We let $\frac{\partial L}{\partial p_{uf}} = 0$, and obtain Equation (8):

$$p_{uf} = \frac{\sum_{(u,i)\in R} \omega \left(1 - C_{ui}\widehat{r^{f}_{ui}}\right) C_{ui} q_{if} - \sum_{(u,i)\notin R} \widehat{r^{f}_{ui}} q_{if}}{\sum_{(u,i)\in R} \omega {C_{ui}}^{2} q_{if}^{2} + \sum_{(u,i)\notin R}\ q_{if}^{2} + \lambda} \tag{8}$$

Observing p_{uf}, we can see that the computational complexity mainly comes from the negatively sampled terms $\sum_{(u,i)\notin R} \widehat{r^{f}_{ui}} q_{if}$ and $\sum_{(u,i)\notin R}\ q_{if}^{2}$. Referring to the eALS, the use of cache matrices in the optimization process can significantly reduce the computational complexity. We set

$$\sum_{(u,i)\notin R} \widehat{r^{f}_{ui}} q_{if} = \sum_{i=1}^{|I|} q_{if} \sum_{k\neq f} p_{uk} q_{ik} - \sum_{(u,i)\in R} \widehat{r^{f}_{ui}} q_{if} = \sum_{k\neq f} p_{uk} \sum_{i=1}^{|I|} q_{if} q_{ik} - \sum_{(u,i)\in R} \widehat{r^{f}_{ui}} q_{if} \tag{9}$$

We define the cache matrix:

$$\mathrm{X}^{q} = \sum_{i=1}^{|I|} q_{if} q_{ik} \tag{10}$$

then we obtain Equation (11):

$$\sum_{(u,i)\notin R} \widehat{r^{f}_{ui}} q_{if} = \sum_{k\neq f} p_{uk} x^{q}_{fk} - \sum_{(u,i)\in R} \widehat{r^{f}_{ui}} q_{if} \tag{11}$$

We substitute the above conclusion into Equation (9) to obtain Equation (12):

$$p_{uf} = \frac{\sum_{(u,i)\in R} (\omega C_{ui} - (\omega {C_{ui}}^{2} - 1)\widehat{r^{\theta}_{ui}}) q_{if} - \sum_{k\neq f} p_{uk} x^{q}_{fk}}{\sum_{(u,i)\in R} (\omega {C_{ui}}^{2} - 1) q_{if}^{2} + x^{q}_{ff} + \lambda} \tag{12}$$

Similarly, We define the cache matrix:

$$\mathrm{X}^{p} = \sum_{u=1}^{|U|} p_{uf} p_{uk} \tag{13}$$

we obtain Equation (14):

$$q_{if} = \frac{\sum_{(u,i)\in R} (\omega C_{ui} - (\omega {C_{ui}}^{2} - 1)\widehat{r^{\theta}_{ui}}) q_{if} - \sum_{k\neq f} q_{ik} x^{p}_{fk}}{\sum_{(u,i)\in R} (\omega {C_{ui}}^{2} - 1) p_{uf}^{2} + x^{p}_{ff} + \lambda} \tag{14}$$

By reusing X^{q} and X^{p}, and optimizing parameters at the element level, which involves optimizing one of the latent vectors while leaving the others fixed, we can ensure that the context-fused model retains the computational complexity of eALS. Algorithm 1 summarizes our method. The whole model optimization process can be easily visualized in Figure 2.

3.4. Context Factor

Aiming at the context factor C_{ij}, which is specific to the paper recommendation system, this paper proposes a feasible method through experiments. The text and citation relationship of the paper itself contains a large amount of relevant a priori information. In this paper, we start with the self-supervised method of text and citation to compute the relevance of users and papers in these two domains. At the same time, in the real engineering environment, this process is carried out offline and does not increase the complexity of the online recommendation process.

Algorithm 1 Context-aware element-wise alternating least squares algorithm

```
Input: $\{r_{ui}\}$: interaction matrix; $\{\omega_{uj}\}$:weight matrix; $D(u_i, v_j)$: user's text preference matrix for papers;
    $N(u_i, v_j)$: user citation preference matrix for papers; $K_d$: latent vector dimension
Output: optimal $\{p\}$: latent vector matrix for users; $\{q\}$: latent vector matrix for papers
Randomly initialize $\{p\}$,$\{q\}$;
repeat
    compute cache matrix for $\{q\}$ $X^q = \sum_{i=1}^{|I|} q_{if} q_{ik}$;
    For u from 1:$|U|$ Do
        For f from 1:K Do
            $p_{uf} \leftarrow (10)$
        END
    END
    compute cache matrix for $\{p\}$ $X^p = \sum_{u=1}^{|U|} p_{uf} p_{uk}$;
    For u from 1:$|I|$ Do
        For f from 1:K Do
            $q_{if} \leftarrow (13)$
        END
    END
until CONVERGE
```

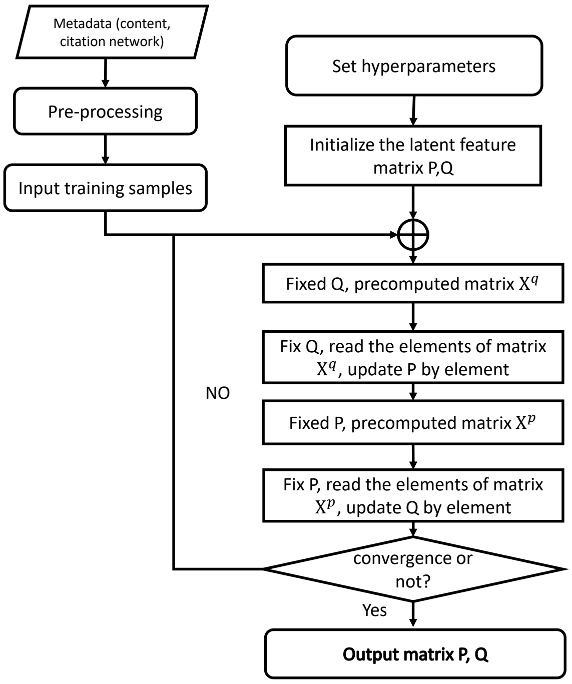
Metadata (content, citation network)
Pre-processing
Input training samples
Set hyperparameters
Initialize the latent feature matrix P,Q
Fixed Q, precomputed matrix X^q
Fix Q, read the elements of matrix X^q, update P by element
NO
Fixed P, precomputed matrix X^p
Fix P, read the elements of matrix X^p, update Q by element
convergence or not?
Yes
Output matrix P, Q

Figure 2. The image shows the optimization of context-aware element-wise alternating least squares, where we use the cache matrix to improve computational efficiency.

3.4.1. Self-Supervised Relevance Modeling Based on Citation Networks

To learn the vector representation of an article within a citation network, we utilize a generative model. Each article, represented as a node in this network, has a low-dimensional vector representation of itself, p, as well as a low-dimensional vector representation of the article when it serves as a context, denoted as p'. Moreover, p' should converge to p. The citation network can be thought of as a directed graph network. The contexts of two article nodes are more closely related and, hence, more relevant, when they have more neighbors. The conditional probability of producing v_j from node v_i for every edge $<i, j>$ in the citation network can be written as follows:

$$P\left(v_j'|v_i\right) = \frac{\exp(p_k'^T * p_i)}{\sum_{k=1}^{|V|} \exp(p_k'^T * p_i)} \tag{15}$$

where $|V|$ is the number of neighbors. Intuitively, two nodes whose context distributions are more similar should be more similar, so the context distributions should approximate their empirical distributions. The empirical distribution $\hat{P}$ can be defined as follows:

$$\hat{P}(v_j|\, v_i) = \frac{1}{d_i} \tag{16}$$

where d_i is the weight of edge $<i, j>$. Here, we choose the degree of node v_i as the value of d_i. We use KL divergence as the objective function to measure the difference between the contextual and empirical distributions. Since the number of negative edges overwhelms the computational power, random negative sampling is introduced into the model computation process to reduce computational effort. Randomized negative sampling involves sampling several negative edges according to the noise distribution for each edge $<i, j>$. In this paper, positive and negative samples are used to optimize the objective function. The objective function can be simplified as follows:

$$O = -\sum_{(i,j)\in E}\left\{\log\sigma\left(p_j'^{T} p_i\right) + \sum_{k=1}^{K}\mathbb{E}_{v_n\sim P_n(v)}\left[\log\sigma\left(p_n'^{T} p_i\right)\right]\right\} \tag{17}$$

where $\sigma(\cdot)$ is the sigmoid function and $P_n(v) \propto (d_v)^{0.75}$. $\mathbb{E}$ is the mathematical expectation. Each time the computational model collects an edge $<i, j>$ from the citation network as a positive sample, it samples K nodes from the noise distribution, $P_n(v)$, to form the negative sample $<i, n>$.

Insufficient training of the related embedding vectors will impair the quality of suggestions because new papers are rarely cited. To address this issue, higher-order neighbors are introduced in this study. Second-order neighbor sampling is utilized for less frequently referenced papers. As illustrated in Figure 3, if the empirical distribution at this moment is expanded as follows for the path $i \to\ k \to\ j$:

$$\hat{P}(v_j|\, v_i) = \hat{P}(v_k|\, v_i) \times \hat{P}(v_j|\, v_k) = \frac{1}{d_i d_k} \tag{18}$$

Finally, we model the citation network correlation score through the above computational process. $N(u_i, v_j) = \sigma\left(\sum_{j=1}^{|N_{u_i}|} p_j'^{T} p_i\right)$, where N_{u_i} is the set of papers in the reading record of user u_i.

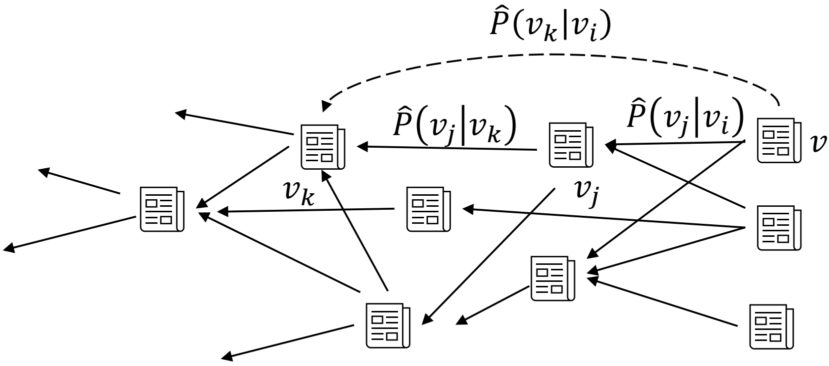

Figure 3. Embedding based on first-order sampling, and for sparse nodes supplemented with the second-order sampling of nodes. When a paper has fewer citations, we collect its second-hop nodes on the citation network as a supplement.

3.4.2. Topic-Based Model for Text Relevance

We use topic modeling to generate textual representations. First, we aggregate the bag of words d_{lj} for any paper l. Similarly, we aggregate all papers in the profile of a given user to form the bag of words d_{ui} for one. Then, we can obtain the bag of words for any user or paper.

Naturally, the topic distributions of papers and readers may be similar. As we can see in Figure 4, the generation of topic distribution is as follows:

For any bag of words d_{ui},

(a) Draw topic proportions $\theta_{d_{u_i}} \sim \mathrm{Dir}(\alpha)$.

(b) For each word $w_{d,n}$ in d_{u_i}:

Draw topic assignment $z_{d,n} \sim \mathrm{Mult}\left(\theta_{d_{u_i}}\right)$

Draw word $w_{d,n} \sim \mathrm{Mult}\left(\phi_{z_{d,n}}\right)$

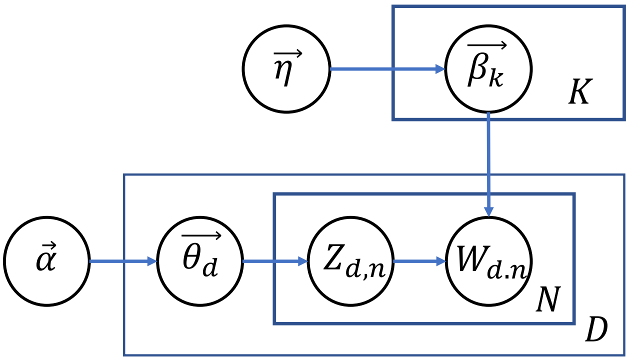

Figure 4. The topic model for users and articles; essentially, each user or paper is represented by a polynomial distribution of the topics.

We use variational inference to estimate the topic–word distribution versus the topic distribution. We use cosine similarity to measure the similarity between two subject distributions. The formula for cosine similarity is as follows:

$$s_c(T_u, T_i) = \frac{T_u * T_i}{||T_u|| * ||T_i||} = \frac{\sum_{k=1}^{N} T_{uk} T_{ik}}{\sqrt{\sum_{k=1}^{N} T_{uk}^2}\sqrt{\sum_{k=1}^{N} T_{ik}^2}} \tag{19}$$

The closer the article is to the users on topic distribution, the closer cosine similarity is to 1.

3.4.3. Contextual Fusion Methods Based on the Multiplication Rule

Based on the multiplication rule [46–48], we integrate these relevance scores into a unified preference score C_{ui}, which is defined as follows:

$$C_{ui} = D(u_i, v_j)N(u_i, v_j) \tag{20}$$

where $D(u_i, v_j)$ is the text relevance between the user topic distribution θ_i and article v_j, as well as the article topic distribution π_j, and $N(u_i, v_j)$ is the citation relevance between user u_i and article v_j. The computation of contextual relevance C_{ui} is performed offline, thus not increasing the computational complexity of the online recommendation process.

3.5. Complexity Analysis

Algorithm 1 shows the optimization of context-aware eALS. Line 3 precomputes the cache matrix X^p according to Equation (10), whose computational complexity is $O(|I|K^2)$, and lines 4–8 compute the user's hidden feature matrix, whose computational complexity is $O(|U|K^2 + |R|K)$. Line 9 precomputes the cache matrix X^q according to Equation (13), whose computational complexity is $O(|U|K^2)$. Lines 10–14 compute the thesis hidden feature matrix q with computational complexity $O(|I|K^2 + |R|K)$. So, the online computational complexity of the whole model is $O((|U| + |I|)K^2 + |R|K)$, proportional to the size of the dataset $|R|$, the size of the set of users $|U|$, the size of the set of papers $|I|$, and the square of the dimension of the hidden features K. It can be seen that the optimization proposed in this paper has the same order of magnitude of complexity as the eALS algorithm [16] without introducing context. Thus, with the preprocessing of the context factor, the computational complexity of the model proposed in this paper remains consistent with the complexity of the eALS algorithm without fusing content information. Therefore, this is currently the more advanced algorithm in terms of computational efficiency. The complexity of these models is listed in Table 2.

Table 2. Time complexity of methods with implicit feedback.

Method	Time Complexity
BPR	$O(\|R\|K)$
PMF	$O(MNK)$
ALS	$O\big((M+N)K^3+\|R\|K^2\big)$
eALS	$O\big((M+N)K^2+\|R\|K\big)$
our method	$O\big((M+N)K^2+\|R\|K\big)$

4. Experiments

This section describes the details of the experiment, including the dataset, parameter settings, metrics, evaluation methods, experimental results, and efficiency analysis.

4.1. Dataset

We used the citeUlike [49] dataset to verify the effectiveness of our method. The citeUlike dataset contains profiles of users, citation networks, titles, abstracts of articles, and tags of articles. The statistical information of the citeUlike dataset is shown in Table 3.

Table 3. Statistics of datasets.

	citeUlike-a	citeUlike-t
Users	5551	7947
Items	16,980	25,975
tags	46,391	52,946
citations	44,709	32,565
User–item pairs	204,987	134,860

4.1.1. Metrics

We recommend K articles to users based on the ranking of predicted values. We choose two metrics to evaluate the quality of the ranked list: NDCG@K and Recall@K, which are defined as follows:

$$NDCG = n_a \sum_j \frac{2^{r(j)}-1}{\log(1+j)} \tag{21}$$

$$\text{Recall} \; = \frac{\sum_u \mathrm{R(u)} \cap T(u)}{T(u)} \tag{22}$$

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 visited by the user.

4.1.2. Baselines

To verify the effect of contextual factors and the optimization method, we selected the following recommended techniques to compare with our own. We chose two memory-based algorithms, userKNN [50] and itemKNN [51], the probabilistic matrix factorization method (PMF) [52], which computes only positive samples, Bayesian personalized ranking (BPR) [15], which is based on stochastic negative sampling, collaborative deep ranking (CDR) [53], and collaborative topic regression (CTR) [54], combining the topic distribution and the implicit behavior of the user, as well as two more advanced context-aware methods: alternating least squares (ALS) [35] and elemental alternating least squares (eALS) [14,16].

We randomly extracted 80% of the data from the user–paper interaction records as training data and the remaining 20% as test data. For a fair comparison, we set the parameters of different algorithms with reference to the corresponding literature.

4.2. Result

4.2.1. Performance Comparison

Tables 4 and 5 show the experimental results of the baseline on the citeUlike-a and citeUlike-t datasets. And we can draw the following conclusions:

Based on two datasets of academic papers, the model proposed in this paper achieves optimal performance in almost all the metrics. This shows that the model has some advantages in implicit feedback-based paper recommendations.

Our models, ALS and eALS, outperform models such as CDR and CTR. Because our models, ALS and eALS, take into account both positive and negative sample sampling, they are somewhat better than models that do not take into account negative samples.

Our method shows more than a 5% improvement over the alternating element multiplier method. We believe this is due to two main reasons: (1) our model uses a more rational approach to compute the trust strength, and (2) our model successfully combines two contexts to mitigate the data sparsity problem. The ALS and eALS models do not model users' implicit feedback well, owing to the extremely sparse data in paper recommendations.

Although the CDR and CTR models explore both implicit user feedback on papers and paper topics, the model proposed in this paper still outperforms the CDR and CTR models. Unlike the CDR and CTR models, which use thesis topics as the bias in their latent vectors, our model employs topic similarity and citation similarity as the context, emphasizing the role of context in enhancing recommendation performance.

Table 4. Performance comparison on citeUlike-a.

	ND@5	**ND@10**	**re@5**	**re@10**
userKNN	0.0235	0.0326	0.0191	0.0429
itemKNN	0.0106	0.0138	0.0067	0.0151
PMF	0.0168	0.0200	0.0151	0.0228
BPR	0.0196	0.0252	0.0199	0.0333
ALS	0.0593	0.0714	0.0408	0.0824
eALS	0.0583	0.0702	**0.0451**	0.0765
CDR	0.0364	0.0442	0.0262	0.0497
CTR	0.0545	0.0637	0.0377	0.0710
ours	**0.0612**	**0.0744**	0.0439	**0.0873**

Superior results are bolded.

Table 5. Performance comparison on citeUlike-t.

	ND@5	**ND@10**	**re@5**	**re@10**
userKNN	0.0226	0.0334	0.0287	0.0585
itemKNN	0.0089	0.0157	0.0130	0.0317
PMF	0.0483	0.0509	0.0542	0.0613
BPR	0.0178	0.0232	0.0167	0.0332
ALS	**0.0510**	0.0598	0.0334	0.0696
eALS	0.0507	0.0614	0.0367	0.0728
CDR	0.0324	0.0411	0.0245	0.0476
CTR	0.0497	0.0547	0.0345	0.0695
ours	0.0507	**0.0634**	**0.0388**	**0.0773**

Superior results are bolded.

In addition, the performance of the BPR model is slightly stronger than that of the PMF model, which is due to the fact that the ratio of positive and negative examples in the two datasets is extremely unbalanced. It also shows that the Bayesian decomposition model is more suitable for modeling the implicit feedback from users on activities. The memory-based approach leads to the worst performance, while the item-based approach is weaker than the user-based approach, which suggests that the sparsity of the items far exceeds the sparsity of the users.

4.2.2. Convergence Analysis

We compared the convergence of four representative methods: BPR, CTR, ALS, and our method. We recorded the NDCG and recall metrics when the models were trained for 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100 epochs.

Figure 5 shows the convergence process of the four models. First, our method performs the best after convergence, confirming that the introduction of contextual information significantly enhances the eALS model's performance. Second, the BPR method achieves convergence before 10 epochs, demonstrating high computational efficiency, but its metrics are not as strong as the other three methods. In addition, there is no significant difference between the convergence of our method and eALS, which confirms the rationality of our method in treating the background information of positive and negative samples differently.

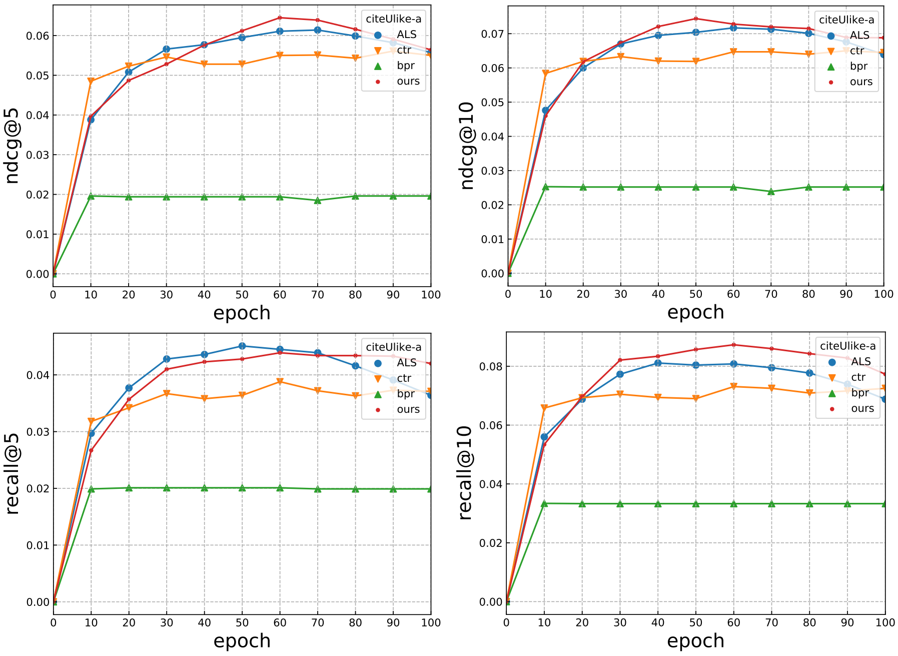

Figure 5. Prediction accuracy of four implicit feedback methods (K_d = 50).

4.2.3. Accuracy vs. Number of Factors K_d

The dimensionality, K_d, of the hidden feature vector is an essential parameter affecting the performance of academic paper recommendations. We progressively increased the value of K_d from 10 to 150 in order to observe the recall and NDCG trends on the citeUlike-a dataset.

The results of the experiment are depicted in Figure 6. As shown in the figure, recall and NDCG values initially increase as K_d increases; however, once the optimal value is reached, recall and NDCG values decrease as K_d increases. On the citeUlike-t dataset, the effect of parameter K_d on recall and NDCG exhibits a similar trend, so we are not going to offer more details. This observation suggests that unduly increasing the value of K_d may introduce noise that reduces the algorithm's precision.

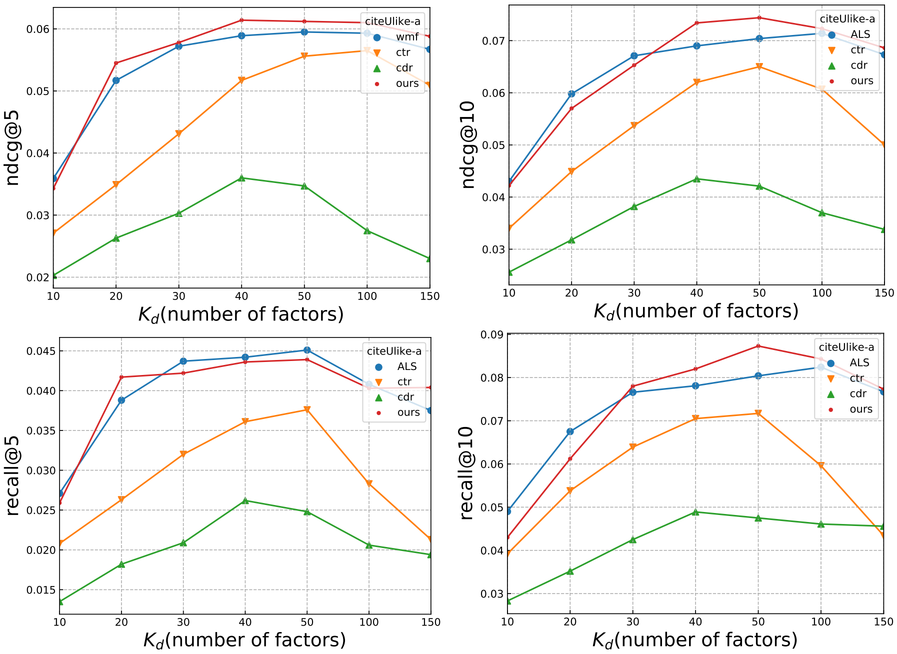

Figure 6. Prediction accuracy of four implicit feedback methods across K_d.

4.2.4. Ablation Experiments

In our method, the parameters $C(u,i)$ are multiplicatively combined based on the relevance scores of the content and citation networks. In this section, we fix the values of content relevance and citation relevance to 1 and compare the changes in NDCG and recall with and without the regularization of the context factor, in order to demonstrate the utility of multiple contextual information.

On the citeUlike-a dataset, Table 6 displays the sensitivity of the metrics to the regularization terms of content relevance and citation relevance. It is evident that citation relevance has the most significant influence on the model.

Table 6. Ablation experiment (citeUlike-a).

Metric	eALS	+Topic	+Citation	Our Method
NDCG@5	0.0593	0.0595	0.0604	0.0612
NDCG@10	0.0714	0.0724	0.0738	0.0744
recall@5	0.0408	0.0425	0.0426	0.0439
recall@10	0.0824	0.0824	0.0858	0.0873

4.2.5. Efficiency

In this part, we compare the running costs across multiple models. The average run time over 30 epochs for many runs on the citeUlike-a dataset is shown in Table 7. For a fair efficiency comparison, all methods were implemented in Python and ran on a single thread on the same computer (Intel i5 8500 3.0 GHz, 8 GB RAM, Geforce 1060 Ti).

We began by contrasting our approach with previous collaborative filtering-based approaches. We discovered that when dimension K rises, each model's running time increases dramatically. Our method's runtime is considerably shorter than that of ALS, which is also based on negative sample full sampling, and is always on the same order of magnitude as eALS, with preprocessed text and citation network similarities. This confirms our estimation of the computational complexity of the models.

Then, we evaluated our approach against other hybrid models. Our approach successfully completed the model's training in a fraction of the time required by the topic regression model CTR and substantially more quickly than the deep learning-based model

CDR. Unexpectedly, we find that CDR converges more quickly when the vector dimension is 50; it will be important to look into this oddity in the future.

Table 7. Training time of 30 iterations for different methods with varying K_d. We compare the running time of models in this section.

K	ALS	eALS	CDR	CTR	Our Method
10	165	25	3485	208	47
20	174	28	4639	232	54
30	192	32	4723	260	61
50	212	42	3773	310	73
100	336	66	5129	490	109
150	433	78	6045	724	148

5. Conclusions and Future Work

Addressing the uncertainty of implicit feedback and the positive and negative sample imbalance in academic paper recommendations, this paper proposes a context-sensitive recommendation method from the perspective of metric learning, which, in addition to taking into account factors such as textual information and citation networks, improves the accuracy of the model while greatly reducing the computational complexity through the introduction of process matrix caching. Experimental results on two real paper recommendation datasets demonstrate the effectiveness of context introduction, with the proposed method showing more than a 5% improvement over the alternating element multiplier method.

In recent years, knowledge graph has shown great potential in recommender systems, and some researchers have combined knowledge graph techniques with traditional collaborative filtering techniques to improve the performance of recommender systems. Combining knowledge graph techniques with the context-sensitive paper recommendation algorithm proposed in this paper will be an interesting research direction.

Author Contributions: Conceptualization, methodology, investigation, original draft preparation, review and editing, W.H.; supervision, B.L.; data curation, Z.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by the National Natural Science Foundation of China, grant no. 62271274.

Data Availability Statement: Data are accessed on 1 January 2020, https://github.com/js05212/citeUlike-a.

Conflicts of Interest: Author Weiming Huang was employed by the company Inner Mongolia Metal Material Research Institute. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Hadhiatma, A.; Azhari, A.; Suyanto, Y. A Scientific Paper Recommendation Framework Based on Multi-Topic Communities and Modified PageRank. *IEEE Access* **2023**, *11*, 25303–25317. [CrossRef]
2. Gündogan, E.; Kaya, M. A novel hybrid paper recommendation system using deep learning. *Scientometrics* **2022**, *127*, 3837–3855. [CrossRef]
3. Kim, D.; Seo, D.; Cho, S.; Kang, P. Multi-co-training for document classification using various document representations: TF–IDF, LDA, and Doc2Vec. *Inf. Sci.* **2019**, *477*, 15–29. [CrossRef]
4. Gündogan, E.; Kaya, M.; Daud, A. Deep learning for journal recommendation system of research papers. *Scientometrics* **2023**, *128*, 461–481. [CrossRef]
5. Wang, G.; Zhang, X.; Wang, H.; Chu, Y.; Shao, Z. Group-Oriented Paper Recommendation With Probabilistic Matrix Factorization and Evidential Reasoning in Scientific Social Network. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 3757–3771. [CrossRef]
6. Ali, Z.; Kefalas, P.; Muhammad, K.; Ali, B.; Imran, M. Deep learning in citation recommendation models survey. *Expert Syst. Appl.* **2020**, *162*, 113790. [CrossRef]

7. Lu, Y.; Yuan, M.; Liu, J.; Chen, M. Research on semantic representation and citation recommendation of scientific papers with multiple semantics fusion. *Scientometrics* **2023**, *128*, 1367–1393. [CrossRef]
8. Mei, X.; Cai, X.; Xu, S.; Li, W.; Pan, S.; Yang, L. Mutually reinforced network embedding: An integrated approach to research paper recommendation. *Expert Syst. Appl.* **2022**, *204*, 117616. [CrossRef]
9. Zhang, J.; Zhu, L. Citation recommendation using semantic representation of cited papers' relations and content. *Expert Syst. Appl.* **2022**, *187*, 115826. [CrossRef]
10. Xiao, X.; Huang, J.; Wang, H.; Zhang, C.; Chen, X. OpenMetaRec: Open-metapath heterogeneous dual attention network for paper recommendation. *Expert Syst. Appl.* **2023**, *231*, 120806. [CrossRef]
11. Xiao, X.; Jin, B.; Zhang, C. Personalized paper recommendation for postgraduates using multi-semantic path fusion. *Appl. Intell.* **2023**, *53*, 9634–9649. [CrossRef]
12. Dong, Y.; Chawla, N.V.; Swami, A. Metapath2vec: Scalable Representation Learning for Heterogeneous Networks. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 13–17 August 2017; KDD '17; pp. 135–144. [CrossRef]
13. Engleitner, N.; Kreiner, W.; Schwarz, N.; Kopetzky, T.; Ehrlinger, L. Knowledge Graph Embeddings for News Article Tag Recommendation. In Proceedings of the Semantics Co-Located Events: Poster&Demo Track and Workshop on Ontology-Driven Conceptual Modelling of Digital Twins co-located with Semantics 2021, Online, 6–9 September 2021; Volume 2941.
14. He, X.; Zhang, H.; Kan, M.Y.; Chua, T.S. Fast Matrix Factorization for Online Recommendation with Implicit Feedback. In Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, Pisa, Italy, 17–21 July 2016; SIGIR '16; pp. 549–558. [CrossRef]
15. Rendle, S.; Freudenthaler, C.; Gantner, Z.; Schmidt-Thieme, L. BPR: Bayesian Personalized Ranking from Implicit Feedback. In Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, 18–21 June 2009; AUAI Press: Arlington, VA, USA, 2009; pp. 452–461.
16. He, X.; Tang, J.; Du, X.; Hong, R.; Ren, T.; Chua, T.S. Fast Matrix Factorization With Nonuniform Weights on Missing Data. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *31*, 2791–2804. [CrossRef]
17. Li, Z.; Zou, X. A Review on Personalized Academic Paper Recommendation. *Comput. Inf. Sci.* **2019**, *12*, 33–43. [CrossRef]
18. Stitini, O.; Kaloun, S.; Bencharef, O. An Improved Recommender System Solution to Mitigate the Over-Specialization Problem Using Genetic Algorithms. *Electronics* **2022**, *11*, 242. [CrossRef]
19. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
20. Ganguly, S.; Pudi, V. Paper2vec: Combining Graph and Text Information for Scientific Paper Representation. In Proceedings of the Advances in Information Retrieval, Aberdeen, UK, 8–13 April 2017; ECIR 2017; Lecture Notes in Computer Science; Springer International Publishing: Cham, Germany, 2017; pp. 383–395. [CrossRef]
21. Chen, L.; Xia, M. A context-aware recommendation approach based on feature selection. *Appl. Intell.* **2021**, *51*, 865–875. [CrossRef]
22. Wu, Y.; Zhao, Y.; Wei, S. Collaborative filtering recommendation algorithm based on interval-valued fuzzy numbers. *Appl. Intell.* **2020**, *50*, 2663–2675. [CrossRef]
23. Hui, B.; Zhang, L.; Zhou, X.; Wen, X.; Nian, Y. Personalized recommendation system based on knowledge embedding and historical behavior. *Appl. Intell.* **2022**, *52*, 954–966. [CrossRef]
24. Sugiyama, K.; Kan, M.Y. Exploiting potential citation papers in scholarly paper recommendation. In Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '13, Indianapolis, IN, USA, 22–26 July 2013; Downie, J.S., McDonald, R.H., Cole, T.W., Sanderson, R., Shipman, F., Eds.; ACM: Indianapolis, IN, USA, 2013; pp. 153–162. [CrossRef]
25. Sun, J.; Ma, J.; Liu, Z.; Miao, Y. Leveraging Content and Connections for Scientific Article Recommendation in Social Computing Contexts. *Comput. J.* **2014**, *57*, 1331–1342. [CrossRef]
26. Wang, G.; He, X.; Ishuga, C.I. HAR-SI: A novel hybrid article recommendation approach integrating with social information in scientific social network. *Knowl.-Based Syst.* **2018**, *148*, 85–99. [CrossRef]
27. Winoto, P.; Tang, T.; McCalla, G. Contexts in a Paper Recommendation System with Collaborative Filtering. *Int. Rev. Res. Open Distance Learn.* **2012**, *13*, 56–75. [CrossRef]
28. Kong, X.; Mao, M.; Wang, W.; Liu, J.; Xu, B. VOPRec: Vector Representation Learning of Papers with Text Information and Structural Identity for Recommendation. *IEEE Trans. Emerg. Top. Comput.* **2021**, *9*, 226–237. [CrossRef]
29. Wang, W.; Tang, T.; Xia, F.; Gong, Z.; Chen, Z.; Liu, H. Collaborative Filtering With Network Representation Learning for Citation Recommendation. *IEEE Trans. Big Data* **2022**, *8*, 1233–1246. [CrossRef]
30. Li, Y.; Wang, R.; Nan, G.; Li, D.; Li, M. A personalized paper recommendation method considering diverse user preferences. *Decis. Support Syst.* **2021**, *146*, 113546. [CrossRef]
31. Zhu, Y.; Lin, Q.; Lu, H.; Shi, K.; Qiu, P.; Niu, Z. Recommending scientific paper via heterogeneous knowledge embedding based attentive recurrent neural networks. *Knowl. Based Syst.* **2021**, *215*, 106744. [CrossRef]
32. Ma, Z.; Xue, J.H.; Leijon, A.; Tan, Z.; Yang, Z.; Guo, J. Decorrelation of Neutral Vector Variables: Theory and Applications. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *29*, 129–143. [CrossRef]
33. Zhang, H.; Shen, F.; Liu, W.; He, X.; Luan, H.; Chua, T.S. Discrete Collaborative Filtering. In Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, Pisa, Italy, 17–21 July 2016; SIGIR '16; pp. 325–334. [CrossRef]

34. He, X.; He, Z.; Du, X.; Chua, T.S. Adversarial Personalized Ranking for Recommendation. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 8–12 July 2018; SIGIR'18; Association for Computing Machinery: New York, NY, USA; pp. 355–364. [CrossRef]
35. Hu, Y.; Koren, Y.; Volinsky, C. Collaborative Filtering for Implicit Feedback Datasets. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 15–19 December 2008; IEEE: Pisa, Italy, 2008; pp. 263–272. [CrossRef]
36. Ding, J.; Yu, G.; He, X.; Quan, Y.; Li, Y.; Chua, T.S.; Jin, D.; Yu, J. Improving Implicit Recommender Systems with View Data. In Proceedings of the International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018; pp. 3343–3349. [CrossRef]
37. Thantharate, P. IntelligentMonitor: Empowering DevOps Environments with Advanced Monitoring and Observability. In Proceedings of the 2023 International Conference on Information Technology (ICIT'23), Amman, Jordan, 9–10 August 2023; pp. 800–805. [CrossRef]
38. Pagano, T.P.; Loureiro, R.B.; Lisboa, F.V.N.; Cruz, G.O.R.; Peixoto, R.M.; Guimarães, G.A.d.S.; Oliveira, E.L.S.; Winkler, I.; Nascimento, E.G.S. Context-Based Patterns in Machine Learning Bias and Fairness Metrics: A Sensitive Attributes-Based Approach. *Big Data Cogn. Comput.* **2023**, *7*, 27. [CrossRef]
39. Hsieh, C.K.; Yang, L.; Cui, Y.; Lin, T.Y.; Belongie, S.; Estrin, D. Collaborative Metric Learning. In Proceedings of the 26th International Conference on World Wide, Perth, Australia, 3–7 April 2017; WWW'17; International World Wide Web Conferences Steering Committee: Perth, Australia, 2017; pp. 193–201. [CrossRef]
40. Tay, Y.; Anh Tuan, L.; Hui, S.C. Latent Relational Metric Learning via Memory-based Attention for Collaborative Ranking. In Proceedings of the 2018 World Wide Web Conference on World Wide Web—WWW '18, Lyon, France, 23–27 April 2018; ACM Press: Lyon, France, 2018; pp. 729–739. [CrossRef]
41. Shrivastava, A.; Gupta, A.; Girshick, R. Training Region-Based Object Detectors with Online Hard Example Mining. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 761–769. [CrossRef]
42. Wang, X.; Han, X.; Huang, W.; Dong, D.; Scott, M.R. Multi-Similarity Loss With General Pair Weighting for Deep Metric Learning. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 5017–5025. [CrossRef]
43. Tran, V.A.; Hennequin, R.; Royo-Letelier, J.; Moussallam, M. Improving Collaborative Metric Learning with Efficient Negative Sampling. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Paris, France, 21–25 July 2019; pp. 1201–1204. [CrossRef]
44. Zhang, S.; Yao, L.; Tay, Y.; Xu, X.; Zhang, X.; Zhu, L. Metric Factorization: Recommendation beyond Matrix Factorization. *arXiv* **2018**, arXiv:1802.04606.
45. Weinberger, K.Q.; Saul, L.K. Distance Metric Learning for Large Margin Nearest Neighbor Classification. In Proceedings of the NIPS, Vancouver, BC, Canada, 5–8 December 2005; pp. 1473–1480.
46. Zhang, J.; Chow, C.Y. 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, Orlando, FL, USA, 5–8 November 2013; pp. 324–333. [CrossRef]
47. Cheng, C.K.; Yang, H.; King, I.; Lyu, M.R. Fused Matrix Factorization with Geographical and Social Influence in Location-Based Social Networks. In Proceedings of the AAAI Conference on Artificial Intelligence, Toronto, ON, Canada, 22–26 July 2012; pp. 17–23. [CrossRef]
48. Liu, B.; Fu, Y.; Yao, Z.; Xiong, H. Learning geographical preferences for point-of-interest recommendation. In Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, IL, USA, 11–14 August 2013; pp. 1043–1051. [CrossRef]
49. Wang, H.; Chen, B.; Li, W.J. Collaborative Topic Regression with Social Regularization for Tag Recommendation. In Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, Beijing, China, 3–9 August 2013; AAAI Press: Beijing, China, 2013; IJCAI '13; pp. 2719–2725.
50. Wang, J.; de Vries, A.P.; Reinders, M.J.T. 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, Seattle, WA, USA, 6–11 August 2006; ACM Press: Seattle, WA, USA, 2006; pp. 501–508. [CrossRef]
51. Sarwar, B.; Karypis, G.; Konstan, J.; Reidl, J. Item-based collaborative filtering recommendation algorithms. In Proceedings of the Tenth International Conference on World Wide Web—WWW '01, Hong Kong, China, 1–5 May 2001; ACM Press: Hong Kong, , China, 2001; pp. 285–295. [CrossRef]
52. Salakhutdinov, R.; Mnih, A. Probabilistic Matrix Factorization. In *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 3–6 December 2007*; Platt, J.C., Koller, D., Singer, Y., Roweis, S.T., Eds.; Curran Associates, Inc.: Vancouver, BC, Canada, 2007; pp. 1257–1264.

53. Ying, H.; Chen, L.; Xiong, Y.; Wu, J. Collaborative Deep Ranking: A Hybrid Pair-Wise Recommendation Algorithm with Implicit Feedback. In Proceedings of the Advances in Knowledge Discovery and Data Mining—20th Pacific-Asia Conference, PAKDD 2016, Auckland, New Zealand, 19–22 April 2016; Lecture Notes in Computer Science; Bailey, J., Khan, L., Washio, T., Dobbie, G., Huang, J.Z., Wang, R., Eds.; Springer: Auckland, New Zealand, 2016; Volume 9652; pp. 555–567. [CrossRef]
54. Wang, C.; Blei, D.M. Collaborative topic modeling for recommending scientific articles. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining—KDD '11, San Diego, CA, USA, 21–24 August 2011; ACM Press: San Diego, CA, USA, 2011; p. 448. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.