Expert Systems With Applications 162 (2020) 113790

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Review

Deep learning in citation recommendation models survey

Zafar Ali [a,*], Pavlos Kefalas [b], Khan Muhammad [c,*], Bahadar Ali [a], Muhammad Imran [d]

[a] School of Computer Science and Engineering, Southeast University, Nanjing, China
[b] Department of Informatics, Aristotle University, Thessaloniki, Greece
[c] Department of Software, Sejong University, Seoul 143-747, Republic of Korea
[d] Information Science and Engineering, Southeast University, Nanjing, China

ARTICLE INFO

Keywords:
Recommender systems
Citation recommendation
Neural networks
Paper recommendation
Machine learning
Deep learning

ABSTRACT

The huge amount of research papers on the web makes finding a relevant manuscript a difficult task. In recent years many models were introduced to support researchers by providing personalized citation recommendations. Moreover, deep learning methods have been employed in this domain to improve the quality of the final recommendations. However, a thorough study that classifies citation recommendation models and examines their (a) strengths and weaknesses, (b) evaluation metrics used, (c) popular datasets, and challenges faced is missing. Therefore, with this survey, we present a new classification approach for deep learning models that provide citation recommendation. Our approach uses the following six criteria: data factors, data representation methods, methodologies, types of recommendations used, problems addressed, and personalization. Additionally, we present a comparative analysis of those models that use the same set of evaluation metrics and datasets. Moreover, we examine hot upcoming issues and solutions in light of explored literature. Also, the survey discusses and analyzes the evaluation metrics and datasets adopted by the explored models. Finally, we conclude our survey with trends and future directions to further assist research on that domain.

1. Introduction

The increasing number of research manuscripts on digital libraries makes a difficult task for a researcher to find similar papers to his/her preferences. To overcome this issue, many models that recommend research papers were introduced to assist users with personalized items or information (Cai, Han & Li, 2018; Khadka & Knoth, 2018; Zhang & Ma, 2020). Typically, citation recommendation systems can been classified into three main approaches, namely: Content-based filtering (CB) (Bhagavatula et al., 2018), Collaborative filtering (CF) (Bansal et al., 2016; Sugiyama & Kan, 2013), and Graph-based (GB) (Christoforidis et al., 2018a; Yang et al., 2018). CB approaches employ the descriptions and features of papers and additional profile information to assist users in producing relevant recommendations. CB models can generate useful recommendations when items descriptions, as well as user preferences, are available, otherwise, such models will encounter the well known cold-start problem (Christoforidis et al., 2018a, 2018b). In contrast, CF-based citation recommendation models exploit the past interactions along with social network to generate recommendations. These models can generate quality recommendations when user feedback and ratings information are available. However, when limited information about users is available, that leads to inaccurate predictions (Son & Kim, 2017) due to sparsity problem. Systems using graph-based models (Cai et al., 2019; Tian & Jing, 2013; Yang et al., 2018) alleviate this issue by exploring auxiliary relationships in the graph. However, traditional GB approaches (Chakraborty et al., 2015; Yang et al., 2018) perceive recommendation as a links prediction task (Son & Kim, 2017).

To overcome these issues, researchers have adopted more sophisticated deep neural networks in producing citation recommendations (Färber et al., 2018a; Yang, Zhang, Cai & Dai, 2019). Deep learning methods are capable of capturing the semantic representations and associated contextual information of research papers, which leads to significant improvement of the final recommendations. Considering the essential role of citation recommendation systems in academia and the emergence of deep learning models, a comprehensive survey in this field is crucial to comprehend in-depth the strengths, weaknesses, application scenarios, problems, and opportunities, along with the evaluation methods in the area.

Motivation There are only four research studies (Ali et al., 2020; Bai et al., 2019; Beel et al., 2016; Ma et al., 2020) in literature that have reviewed the domain of citation recommendation. A comparative analysis of these surveys and our proposed survey has been given in Table 1.

* Corresponding authors.
E-mail addresses: zafarali@seu.edu.cn, zafaralibagh6@gmial.com (Z. Ali), kefalasp@csd.auth.gr (P. Kefalas), khan.muhammad@ieee.org (K. Muhammad), write2bahadar@gmail.com (B. Ali), imranafkary@seu.edu.cn (M. Imran).

https://doi.org/10.1016/j.eswa.2020.113790
Received 28 February 2020; Received in revised form 21 July 2020; Accepted 21 July 2020
Available online 24 July 2020
0957-4174/

Table 1
Comparison with other survey papers.

Survey Ref.	Coverage	Models types	Comparative analysis	Issues and solutions	Trends	Key findings/limitations
(Beel et al., 2016)	1998–2013	General	Not provided	Issues and solutions	Derived	*Categorization based on CB, CF and hybrid. *Does not provide comparative analysis among the explored models. *Does not cover new algorithms and datasets
(Bai et al., 2019)	2006–2018	General	Not provided	Only few issues	Not derived	*Classification based on CF, CB and Hybrid. *Does not cover datasets and their comparision. *Does not provide comparative analysis among the explored models. *Limited number of newly proposed models explored
(Ma et al., 2020)	2008–2018	General	Not provided	Only few issues	Not derived	*Classification using query-based and information filtering methods. *Does not cover comparison among the datasets. *Does not provide comparative analysis among the explored models
(Ali et al., 2020)	2010–2020	General	Not provided	Issues and solutions	Derived	*Classification based on CB, CF, and hybrid. *Covers comparison among the datasets. *Does not provide comparative analysis among the explored models
This survey	2010–2020	Deep learning based	Provided	Issues and solutions	Derived	*Classification using a novel taxonomy of six criteria. *Local and global citation recommendation. *Personalized and non-personalized recommendation. *Comparison among the datasets

In this connection, Beel et al. (2016) classified citation recommendation models by using three information filtering methods such as CF, CB, and Hybrid. This survey covered the literature which is published until 2013 and discussed different citation recommendation models, evaluation datasets, metrics, and future research directions. However, due to the rise of the novel recommendation methods and algorithms such as the ones using deep neural networks and representation learning methods in the last seven years, a new comprehensive framework and taxonomy are needed. Additionally, the survey is limited in terms of presenting a comparative analysis of the explored models. In the same direction, Bai et al. (2019) and Ali et al. (2020) surveyed the domain by classifying the recommendation models using information filtering methods such as collaborative filtering, content-based filtering, and hybrid. Additionally, these surveys presented commonly used metrics and open research issues but lack in exploiting the novel recommendation methods and presenting a comparative analysis among them. Additionally, the survey (Bai et al., 2019) does not cover commonly used evaluation datasets, and comparative analysis among these datasets. Furthermore, the survey is limited in terms of presenting implications and future directions. On the other hand, a recent survey (Ma et al., 2020) classifies citation recommendation models based on user queries, namely: keyword-based, citation list-based, and context-based citation recommendations. Additionally, the survey report commonly used evaluation metrics in the domain. However, like the aforementioned surveys, it does not cover the data factors/features exploited, data representation techniques adopted, and more advanced recommendation approaches employed by deep learning-based citation recommendation models. Moreover, it does not provide a comparative analysis among the algorithms and datasets adopted in the domain. Finally, the survey is limited in terms of exploring the prominent issues and their solutions, which are related to the domain of citation recommendation.

Due to the advent of novel research models that employ deep learning and network embedding methods, an inclusive study is important. To this point this survey explores the citation recommendation domain and in particular we focus on models that use deep neural networks.

Contributions The contribution of this survey is summarized into the following points:

- This study surveys 35 citation recommendation models using deep neural networks, published during the last decade. We categorize these models using the following six criteria: (a) data factors/information employed, (b) data representation methods, (c) methodologies adopted, (d) recommendation types, (e) problems faced, and (f) personalization. To the best of our knowledge, this is the first thorough survey that focuses on deep learning-based citation recommendation models.
- Also, we examine the most popular evaluation metrics and datasets used, and present a comparative analysis among them. This makes dataset selection easier for research community, considering their preferences and the domain of application.
- Moreover, we compare and analyze the experimental results of different citation recommendation models that use the same set of metrics and evaluation datasets.
- Furthermore, we present prominent research issues and solutions in the light of new trends, implications, and directions for future research on the domain.

This paper is organized in such a way that Section 2 reviews 35 citation recommendation models that use deep learning methods and classifies them considering multiple criteria. In Section 3, we analyze the evaluation datasets and metrics used along with a comparison among the datasets. Section 4 presents a comparative analysis of citation recommendation models based on their experimental results over three evaluation datasets. Section 5 discusses prominent research problems and challenges related to the citation recommendations. Finally, Section 6 concludes the survey.

2. Recommendation models classification

In this section, we first examine and then we categorize 35 citation recommendation models that employ deep neural networks extending previous work (Kefalas et al., 2015). We classify citation recommendation models using the below six criteria: (1) data factors, (2) representation of the data, (3) methodologies adopted, (4) recommendation types provided, (5) problems addressed, and (6) the penalization type as shown in Table 2 and Fig. 1. The proposed multilevel taxonomy aims to examine the strengths and weaknesses of the explored models and propose domain trends. In the next sections, we cover a detailed overview of the contents of the Table 2.

Table 2
Classification of citation recommendation models.

		Data factors/Information used							Data representations			Methodologies adopted							Recommendation types			Problem faced		Personalization	
	Model	Paper contents	Tags/keywords	User/author profile	Venue information	Citation network	Social network	Ratings	Matrix-based	Graph-based	Hybrid	Embeddings	RBM	MLP	GAN	CNN	RNN	DRL	Local	Global	Tags/Labels	Sparsity	Cold-Start	Non-Personalized	Personalized
1	CIRec (Chen et al., 2019b)	✓	✓	–	–	✓	–	–	–	✓	✓	✓	–	–	–	–	–	–	✓	–	–	–	–	–	✓
2	RBM-CS (Tang & Zhang, 2009)	–	–	✓	–	✓	–	–	–	✓	–	–	✓	–	–	–	–	–	✓	–	–	–	–	–	✓
3	MMRQ (Mu et al., 2018)	✓	✓	✓	–	✓	–	–	–	✓	–	–	–	–	–	–	–	✓	–	✓	–	–	–	–	✓
4	MAAE (Galke et al., 2018)	✓	✓	✓	–	✓	–	✓	✓	–	–	–	–	–	✓	–	–	–	–	✓	✓	✓	–	–	✓
5	HGRec (Ma & Wang, 2019)	✓	✓	✓	✓	✓	✓	–	–	✓	–	✓	–	–	–	–	–	–	–	✓	–	–	✓	–	✓
6	POLAR (Du et al., 2019)	✓	–	✓	–	–	–	–	✓	–	–	–	–	–	–	✓	–	✓	–	✓	–	–	–	–	✓
7	RI-PR (Bulut et al., 2020)	✓	✓	✓	–	–	–	–	–	–	✓	✓	–	–	–	–	–	–	–	✓	–	–	–	–	✓
8	DocCit2Vec (Zhang & Ma, 2020)	✓	–	–	–	✓	–	–	–	–	✓	✓	–	–	–	–	–	–	✓	–	–	–	–	✓	–
9	DRDF-CR (Kobayashi et al., 2018)	–	–	✓	–	✓	–	–	–	✓	–	–	–	✓	–	–	–	–	✓	–	–	–	–	–	✓
10	HRLHG (Jiang et al., 2018)	✓	✓	✓	–	✓	–	–	–	✓	–	✓	–	–	–	–	–	–	–	✓	–	–	–	–	✓
11	ML-DTR (Bansal et al., 2016)	✓	✓	✓	–	–	–	✓	✓	–	–	–	–	–	–	–	✓	–	–	✓	–	✓	✓	–	✓
12	SAR (Gupta & Varma, 2017)	–	–	✓	–	✓	–	–	–	✓	–	✓	–	–	–	–	–	–	–	✓	–	–	–	–	✓
13	BERT-GCN (Jeong et al., 2019)	✓	–	✓	–	✓	–	–	–	✓	–	–	–	–	–	✓	–	–	✓	–	–	–	–	–	✓
14	NCN (Ebesu & Fang, 2017)	✓	–	–	–	✓	✓	–	–	–	✓	–	–	–	–	✓	✓	–	✓	–	–	–	–	–	✓
15	PPR-DL (Hassan, 2017)	✓	–	✓	–	–	–	–	–	–	–	–	–	–	–	–	✓	–	–	✓	–	–	–	–	✓
16	ASL (Dai et al., 2019)	✓	–	✓	–	✓	–	–	–	✓	✓	–	–	–	–	–	✓	–	✓	–	–	–	–	–	✓
17	CITEWERTs(a)(Färber et al., 2018b)	✓	–	–	✓	–	–	–	–	–	–	–	–	–	–	✓	✓	–	✓	–	–	–	–	✓	–
18	CITEWERTs(b)(Färber et al., 2018a)	✓	–	–	✓	–	–	–	–	–	–	–	–	–	–	✓	✓	–	✓	–	–	–	–	✓	–
19	WHIN-CSL (Chen et al., 2019a)	✓	–	✓	✓	✓	✓	–	–	✓	–	✓	–	–	–	✓	✓	–	–	✓	–	–	–	–	✓
20	NNRank (Bhagavatula et al., 2018)	✓	–	✓	✓	–	✓	–	–	–	–	✓	–	–	–	–	–	–	–	✓	–	–	–	–	✓
21	GAN-HBNR (Cai, Han & Yang, 2018)	✓	–	✓	–	✓	✓	–	–	✓	–	–	–	–	✓	–	–	–	–	✓	–	✓	–	–	✓
22	PCCR (Yang et al., 2018)	✓	–	✓	–	✓	–	–	–	✓	–	–	–	–	–	–	✓	–	✓	–	–	–	–	–	✓
23	NPM (Huang et al., 2015)	✓	✓	✓	–	–	–	–	–	–	✓	–	–	✓	–	–	–	–	–	✓	–	–	–	–	✓
24	Paper2vec (Tian & Zhuo, 2017)	–	–	–	–	✓	–	–	✓	–	–	✓	–	–	–	–	–	–	–	✓	–	–	–	✓	–
25	VOPRec (Kong et al., 2019)	–	–	✓	–	✓	–	–	✓	✓	–	✓	–	–	–	–	–	–	–	✓	–	–	–	–	✓
26	p-CNN (Yin et al., 2017)	✓	–	✓	–	–	✓	–	–	–	✓	–	–	–	–	✓	–	–	✓	–	–	–	–	–	✓
27	VCGAN (Zhang et al., 2018)	✓	–	✓	–	–	✓	–	–	–	✓	–	–	–	✓	–	–	–	–	✓	–	–	–	–	✓
28	TMR-PCR (Cai, Han & Li, 2018)	–	–	✓	–	✓	–	–	✓	✓	–	–	–	–	–	–	–	✓	–	✓	–	–	–	–	✓
29	CPR (Sharma et al., 2017)	✓	–	✓	–	–	–	–	–	–	✓	✓	–	–	–	–	–	–	–	✓	–	–	–	–	✓
30	BNR (Cai et al., 2019)	✓	–	✓	–	✓	–	–	✓	✓	–	✓	–	–	–	–	–	–	–	✓	–	–	–	–	✓
31	HRM (Li et al., 2019)	✓	✓	✓	✓	–	✓	–	✓	–	–	–	–	✓	–	–	–	–	–	✓	–	✓	✓	–	✓
32	CPA-CE (Khadka & Knoth, 2018)	✓	–	–	–	✓	–	–	–	–	✓	✓	–	–	–	–	–	–	–	✓	–	–	–	–	✓
33	AED (Yang, Zhang, Cai & Dai, 2019)	✓	–	–	–	✓	✓	–	–	–	✓	–	–	–	–	✓	✓	–	✓	–	–	–	–	–	✓
34	NREP (Yang, Zhang, Cai & Guo, 2019)	–	–	✓	–	✓	–	–	–	✓	✓	✓	–	–	–	–	–	–	–	✓	–	–	–	–	✓
35	HIPRec (Ma et al., 2019)	–	✓	✓	✓	✓	–	–	–	✓	–	✓	–	–	–	–	–	–	–	✓	–	–	✓	–	✓

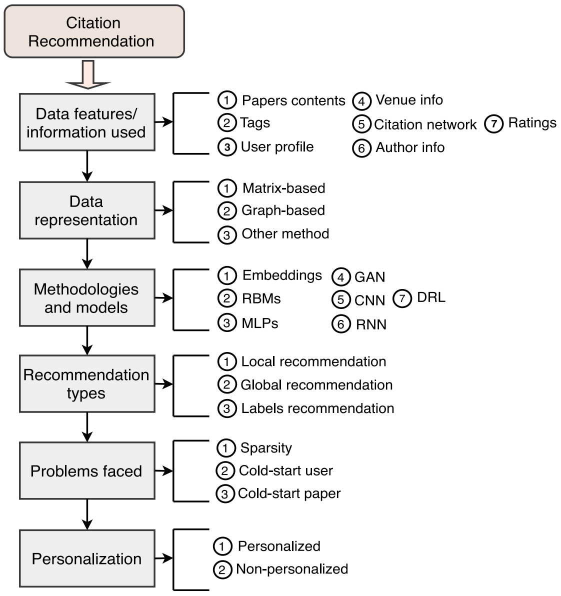


Fig. 1. An overview of algorithms taxonomy.

2.1. Data factors/features

Citation recommendation models employ different factors and features including: paper content, tags/keywords, user profile, venue information, citations network, author information, and ratings while recommending relevant citations to users. Below, we examine the explored models with respect to the aforementioned factors employed in producing citation recommendations.

Papers content: A research paper consists of the abstract, the title, the metadata, and the logical components if available such as the introduction, the methodology, the experimental results, etc. Researchers want a system that recommends articles with content close to their research preferences and interests. Therefore, we can observe in column 3 of Table 2 that the majority of the explored models exploit such content.

Tags/keywords: is another important feature since it correlates keywords and papers. A research manuscript has a set of keywords/tags that provide a brief description of its contents. By using tags, a CB system correlates a target user's interests with similar articles. It is worth-mentioning that many of the explored models support this feature. On the other hand, it is noticeable that models ignoring tags exploit users' profiles, ratings, or auxiliary networks (Xia et al., 2016; Zhang & Ma, 2020).

User profile: User profile is considered the most popular among the rest features used as it provides rich information about past preferences while finding researchers with similar research interests. For instance, users' interaction list, such as ratings, downloads, read etc. may exploit their interests and refine the generated recommendations.

Venue information: To generate adequate citation recommendations, some models employ users' preferences related to venues (Cai et al., 2019). A venue that publishes papers related to the query manuscript is of great importance to the researcher than a random venue. Thus venue information is expected to boost the quality of the recommendations. In literature, we can notice in Table 2 that only six models exploited that aspect.

Citation network: Citation network reveal a strong relationship and bond among the papers. In a citation network, we consider papers as nodes and references from one paper to another is denoted with an edge. This network contains rich information about the relations with the majority of the models using this side information.

Social network: Authors tend to follow, collaborate, or been friends with other researchers with similar interests. Such kind of bonds may have a huge impact on the final recommendations (Ebesu & Fang, 2017). Thus, there is a higher probability of a user being interested in a research paper published from authors in his/her social network rather than a random author in publications libraries. To this point, multiple models adopted this feature to further refine recommendations.

Ratings: Citation recommendation models use users' ratings to personalize recommendations. In particular, researchers rate papers in a scale of 1–5 stars. These models aim to identify the new papers that are similar to users' past preferences. Unfortunately, this information is related to embedded systems thus is quite rare to find models using this feature. In the literature, only two models (Bansal et al., 2016; Galke et al., 2018) exploit users' ratings.

2.2. Data representation methods

This section covers various data representation techniques employed in explored studies.

Matrix-based: is referred to algorithms that use a matrix for representing data. In particular, we have a user–item rating matrix with r_{ij} denoting that a user i interacted with item j. That is, it considers the implicit feedback setting and observes that whether a person has interacted with an item. The contents of this matrix may be 1 if user interacted with the particular item or 0 otherwise. Similarly, that matrix may contain users' ratings. The model utilize the items R_i^+ that the user interacted with to suggest top K relevant items from unseen items i.e., R_i^-. On this direction, Multi-model Adversarial Auto-encoder (MAAE) (Galke et al., 2018) uses a ratings matrix $X \in (0, 1)$ showing implicit feedback deriving from a document j over another node k. In contrast to traditional user–item rating matrix $U \times I$, here the research papers are considered as users over their authors.

The rationale is that an author can have collaboration in multiple research works related to various domains, but that all authors concerning the given paper should get similar recommendations. Similarly, a research paper should receive the same set of recommendation results for candidate subjects. In literature, only few models (Bansal et al., 2016; Cai, Han & Li, 2018; Cai et al., 2019; Du et al., 2019; Galke et al., 2018; Kong et al., 2019; Li et al., 2019; Tian & Zhuo, 2017) have adopted matrix for data representation.

Graph-based: employ k-partite graphs to explore useful relations among nodes. In this direction, Bibliographic Network Representation (BNR) (Cai et al., 2019) employs graph structure along with the content, i.e., authors, papers, and venues, to generate low-dimensional representations of objects. Then, the model generates citation suggestions by computing the similarity of the papers and authors representations. In Fig. 1 under column "Data representations", we can see that 17 models have used graphs to represent data.

Hybrid: consists a mixture of other approaches such as graph and matrix combined or other alternative ways. The increasing number of models (Bulut et al., 2020; Chen et al., 2019a; Dai et al., 2019; Ebesu & Fang, 2017; Huang et al., 2015; Khadka & Knoth, 2018; Yang, Zhang, Cai & Dai, 2019; Yang, Zhang, Cai & Guo, 2019; Yin et al., 2017; Zhang et al., 2018) using this kind of representation indicates a new trend in the domain.

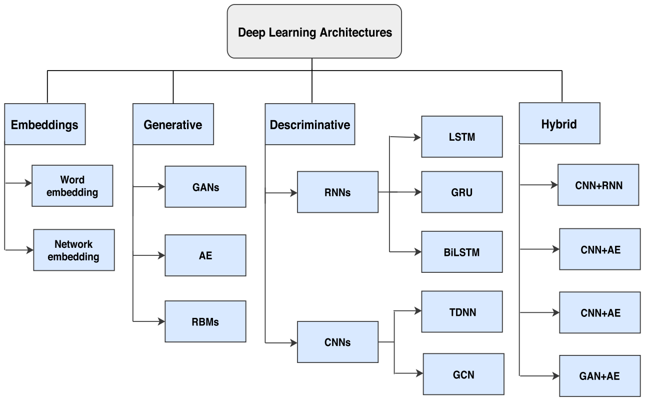


Fig. 2. Deep Learning Architectures.

2.3. Methodologies and models

In recent years, the Artificial Neural Networks (ANN) gained significant results in the domain (Batmaz et al., 2019). Similar to Restricted Boltzmann Machine (RBMs) (Jordan et al., 2001), embedding methods (Grover & Leskovec, 2016; Mikolov et al., 2013), Convolutional Neural Network (CNN) (Liu et al., 2017), and Recurrent Neural Network (RNN) (Goodfellow et al., 2016), ANNs are used to learn a complex mapping of the network. Below, we discuss the different types of ANN approaches adopted by systems that provide citation recommendation as depicted under column *methodologies and models* of Table 2.

Generally, an ANN comprises various layers, such as an input, hidden layer/s, and an output layer where each layer possesses a set of neurons (Liu et al., 2017). This section covers an overview of the DL architectures that are closely related to the explored models as depicted in Fig. 2.

Restricted Boltzmann Machine (RBM): The constituent units are connected symmetrically that make stochastic decisions about whether to be on or off. Besides, these neurons can have intra-layer connectivity as well. Additionally, its learning algorithm is very simple and slow when the network has many layers. In contrast, Restricted Boltzmann Machine (RBM) (Jordan et al., 2001) is a two-layer NN, which contains just an input and a hidden layer. In BMs, connections between layer are restricted, therefore the implementation of RBMs is easier than Boltzmann Machines. Compared to BMs, the learning efficiency of RBMs is significantly improved due to the restrictions on the intra-layer connectivity between the neurons (Mu, 2018). A basic form of an RBM network is depicted in Fig. 3).

In the hidden layer H_i, we compute the score for each node by multiplying four inputs of visible layer V_l with their weights. Then, the sum of those products is added to a bias. Finally, an activation function pass the results to the output as $h = g(W * v + b)$. The probability to observe a certain state of v and h is computed as:

$$p(v, h) = \frac{1}{Z} e^{-(a^T v + b^T h + h^T w v)} \tag{1}$$

where a and b are the biases of both the visible and hidden layers, respectively. Z denotes the normalized function, while W is used for the weight parameter between the V_l and H_l. This is done by optimizing the following objective:

$$arg\ max \sum_{v \in V} log\ P(v, h) \tag{2}$$

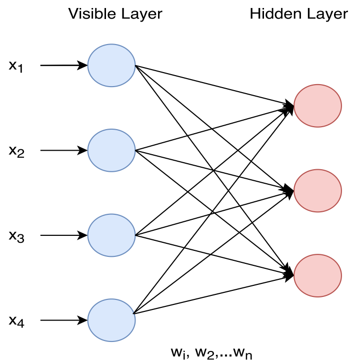


Fig. 3. Architecture of Restricted Boltzmann machine.

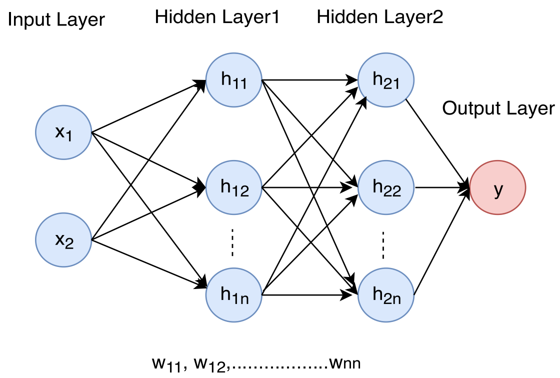


Fig. 4. An example of MLP network.

In contrast to other networks like sparse Autoencoder, the performance of RBM is very fast as it requires simple forward encoding. To this end, Tang and Zhang (2009) proposed Restricted Boltzmann Machine (RBM) with two-layers called RBM-CS. They learn the topic distribution of the article's contents and citation ties. That is, it captures the topic distribution of papers with a hidden topic layer, which is conditioned by utilizing these two relationships to generate a ranked list of top-n papers.

Multi Layer Perceptron (MLP): is considered to be the simplest type of a NN, which is also known as feed-forward method. In particular, we have one or more than one hidden layer/s and one output layer (Zhang et al., 2019). These layers contain activation functions simply by tuning the weights while training. When the hidden layer/s receive the input representation, they map input and output by applying non-linearity (Liu et al., 2017) as depicted in Fig. 4. The output of each neuron is computed using $h = g(W * x_i + b)$. Here, the W represents weights between two neurons of corresponding input and hidden layers. While h and b denotes hidden layer neuron and the bias vector, respectively, g is used as activation functions. Among the most known is relu, tanh, sigmoid, etc. Finally, the output of the network is predicted by propagating the weighted sums of hidden layer/s to the output layer, which applies non-linearity to the transmitted value to produce the final prediction as follows:

$$\hat{y} = g(W * h + b) \tag{3}$$

where $\hat{y}$ denotes an output and g represents the non-linear activation function. Non-linear activation functions are used to propagate to the next layer.

In citation recommendation, MLPs have shown promising results in capturing the semantic representations of papers and producing quality results. To this point, Bhagavatula et al. (2018) introduced NNRank which is a three layer feed-forward neural model that encodes research papers into a low-dimensional space employing the content of research papers. Then it identifies K nearest neighbors for the query paper and employs another discriminative model between observed and unseen papers to re-rank the papers. The model is trained to predict a high cosine similarity for the pair (d_q, d^+) and a low cosine similarity for the pair (d_q, d^-) using the per-instance triplet loss as defined:

$$Loss = max(\alpha + s(d_q; d^-) - s(d_q; d^+), 0) \tag{4}$$

where $s(d_i, d_j)$ is defined as the cosine similarity between document embeddings $cos - sim(e_{d_i}, e_{d_j})$. Where α is used as a hyperparameter of the model. The model generates significant results compared to other models, especially when metadata (i.e., author information, venue information, key-phrases, etc.) is not available. Similarly, Huang et al. (2015) employs the semantic representations of references and their contexts to produce recommendations. That is, a multi-layer NN is trained to learn the probability of references given the contexts, this way it computes a score for each paper d_j as follows:

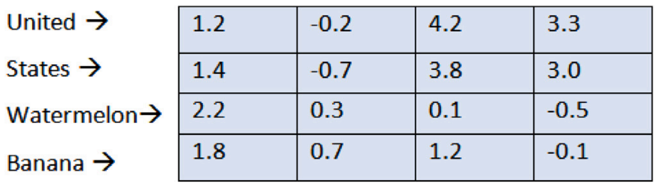
$$p(d_i \mid q) = \sum_{j=1}^{q} p(d_i \mid w_j) p(w_j \mid q) \tag{5}$$

where $p(w_j \mid q)$ denotes the probability of an article given a query q. In contrast, Kobayashi et al. (2018) introduced a CB co-citation model called DRDF-CR, which learns multi-vector representations of the text along with the citation graphs. In particular, each vector captures the discourse facets of a paper, capable of producing context-aware recommendations. To classify sections of an article, the model uses the fastText library (Bojanowski et al., 2017). In particular, it first computes the vector representations of each section in an article by taking vectors average of all words that exist in the section. Then it computes the facet representations (i.e., objective, method, and result) of a paper by taking the average of all facets correspond to that paper. Next, the citation graph is augmented by adding a discourse facet to each citation edge. Finally, the model updates each facet/section vector by employing LINE (Tang et al., 2015) model.

Similarly, Li et al. (2019) introduced a model named Hybrid Reranking Model (HRM) that employs a two-layer feedforward NN to generate predictions, where the input is articles features, and in the output layer a prediction score is generated for each article. The model emails weekly article recommendations to registered users. More specifically, it exploits the content of articles and user behaviors to re-rank the proposed articles generated by the ScienceDirect. That is, it exploits different content-based measures which are extracted using different aspects including: space, tags, author similarity. Next, the joint MF is employed to map user's browsed articles to a click made by the user. Additionally, the model uses a pairwise learning approach to re-rank the final list.

Embedding methods: encodes discrete variables into a low-dimensional, learned continuous vector representations. These embeddings mitigate the shortcomings of traditional encoding methods (i.e., one-hot encoding) and can be used for different purposes including: (1) identifying nearest neighbors, which can be used to make recommendations based on user/group interests, (2) input into another neural network, and (3) in visualizing the concepts and relations between different categories. Embedding techniques can be broadly classified into word embedding (Mikolov et al., 2013) and graph embedding (Cui et al., 2019) methods. In NLP, word embedding is used for representing a set of phrases and feature learning methods where words from the vocabulary are encoded into vectors. Such methods have been tested to capture robust syntactic and semantic information on words. The most popular models are BERT (Devlin et al., 2018) word2vec (Mikolov et al., 2013), and doc2vec (Le & Mikolov, 2014) to embed users, items, documents and locations (Christoforidis et al., 2018b) into a latent space.

For instance, the word 'United' as an input to an embedding model, it is most likely that the model will generate its representation similar



United →	1.2	-0.2	4.2	3.3
States →	1.4	-0.7	3.8	3.0
Watermelon→	2.2	0.3	0.1	-0.5
Banana →	1.8	0.7	1.2	-0.1

Fig. 5. A 4-dimensional vector representation of words.

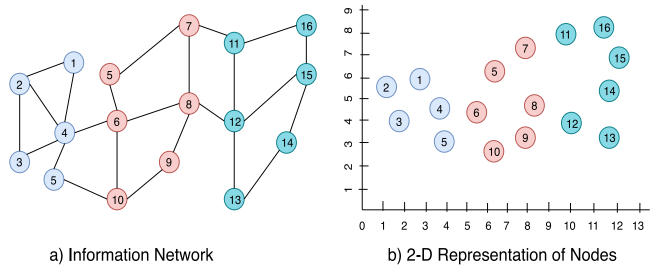


Fig. 6. An Illustration of Network representation learning.

to the related word like ‘States’ than the words such as ‘Banana’ and ‘Watermelon’, which are unrelated. To do so, the model analyzes the contexts (i.e., what words are likely to appear around the words) of these words and generate their vector representations as depicted in Fig. 5. Thus, the 4-dimensional representation reveals the words which occur in similar contexts, possess similar representations in the embedding space.

In contrast, graph embedding encode the nodes into low-dimensional space by exploiting the topological structure, along with other auxiliary information such as contents, attributes etc. Such representations are used to capture the relationships based on computing the distances of the nodes in the embedded space as depicted in Fig. 6.

Notice that, the proximate vertices are kept closed in the latent space. For instance, node 7 possess first-order proximity with node 11, therefore they are closer to each other. Though nodes 12 and 14 are not directly linked but they share common neighbors and therefore maintain second-order proximity, thus they are kept closed in the embedding space. In literature, different learning methods such as LINE (Tang et al., 2015), DeepWalk (Perozzi et al., 2014), HINE (Shi et al., 2019) and Node2vec (Grover & Leskovec, 2016) introduced. Notice that, these approaches are classified into homogeneous and heterogeneous graph embeddings. In homogeneous network embedding models (i.e., LINE, DeepWalk etc.), only single type of nodes and relations are exploited, such methods cannot handle heterogeneity. In contrast, heterogeneous network embedding models (i.e., HINE, Node2vec, etc.) explore multiple kinds of nodes and relations while generating vector representations. A detailed discussion on novel network embedding approaches can be found in Cui et al. (2019).

In the context of citation recommendations, the aforementioned embedding methods are used either to find similarity between nodes or as an input to other (i.e., typically a supervised learning) methods that provide the recommendations. In this direction, Chen et al. (2019b) proposed a weighted heterogeneous network embedding model called Citation Tendency Random Walk (CIRec). More specifically, it captures relations between papers and their references by employing a weighted version of random walk. In particular, the citation tendency is computed by exploiting the ties between papers and their citations using common features such as authors and terms. Then, it employs the skip-gram to obtain the vector representations of research papers, and optimize the following objective function:

$$L = \sum_{v_i \in V} log Pr(N_s(v_i) \mid v_j) \tag{6}$$

where V represents all the nodes. $N_s(.)$ contains the neighborhood information on the same sliding window with v_i. Finally, it uses cosine similarity to find similarity between papers in the embedding space and then it generates recommendations. In contrast, Chen et al. (2019a) introduced WHIN-CSL that exploits the semantic relations and attribute values on relations to generate reference recommendations. In particular, it constructs a weighted HIN that contain papers and authors nodes and exploiting four different relations such as writing, semantic linking, citing, and co-author. Then, it employed Node2vec (Grover & Leskovec, 2016) to obtain vector representations of nodes. Finally, the model computes similarity between vertexes and employs a linear combination method such as:

$$\begin{aligned} P_r(cp_c \mid tp_t) = {} & w_1\mu_1(cp_c, tp_t) + w_2\mu_2(cp_c, tp_t) \\ & +(1 - w_1 - w_2)\mu_3(cp_c, tp_t) \end{aligned} \tag{7}$$

where $P_r(cp_c \mid tp_t)$ denotes the conditional probability between target paper tp_t and candidate paper cp_c. Also, $\mu_1(cp_c, tp_t)$, $\mu_2(cp_c, tp_t)$ and $\mu_3(cp_c, tp_t)$ represents abstract–abstract similarity, paper–paper vertex similarity, and paper–author vertex similarity, respectively. The $w_1 + w_2 < 1$ are employed to tune the final results of each relation. This way, the model generates top-n relevant reference recommendations.

On the other hand, Ganguly and Pudi (2017) introduced Paper2vec that exploits both vertex content and network structure. The model uses an unsupervised neural embedding method to generate paper embeddings and consequently citation recommendations. To learn papers representations in the network G, the model employs Skip-gram and CBOW along with negative sampling. The results reveal that it outperformed baseline embedding models like DeepWalk and Doc2vec in producing quality recommendations. On the other hand, Ma and Wang (2019) exploits both the content and network structure to generate personalized citation recommendations with a model called HGRec. This model generates the author and paper profiles along with the node vectors by exploiting the contents using Doc2vec. Then, it updates jointly the node embeddings by employing two meta-path first and second-order proximity. Finally, the similarity between paper and author embedding vectors are computed to generate citation recommendations.

Another model named VOPRec (Kong et al., 2019) exploits the textual information along with network structure to learn vector representations. The model employs Paper2vec and Struct2vec emebdding techniques to learn text-based and structure-based vectors, respectively, which are used to produce final recommendations. In the same direction, NREP (Yang, Zhang, Cai & Guo, 2019) incorporates the contents, and network structure to learn nodes representations, and produces relevant citation recommendations. To do so, the model adopts a joint learning method that minimizes the following objective function:

$$minL = min(\delta L_c + \theta L_0 + \gamma L_h) \tag{8}$$

where L_c, L_0 and L_h represents the objective functions of vector-content correlation, network structure-based proximity and edge prediction objective functions, respectively. While δ, θ and γ hyperparameters are used to tune the importance of the three objective functions. The easy application and the benefits of the embedding information into the same latent space makes this approach one of the most popular (Bhagavatula et al., 2018; Bulut et al., 2020; Cai et al., 2019; Chen et al., 2019a; Gupta & Varma, 2017; Jiang et al., 2018; Khadka & Knoth, 2018; Kong et al., 2019; Ma & Wang, 2019; Ma et al., 2019; Tian & Zhuo, 2017; Yang, Zhang, Cai & Guo, 2019).

Convolutional Neural Network (CNN): are close to MLP, which require less pre-processing and perform well on largescale networks with low memory consumption while training. Their structure consists an input, a convolutional, a sub-sampling layer that performs the pooling, a fully connected layer, and an output layer, as depicted in Fig. 7. This structure generated a feature map f^k, which is computed as follows:

$$f_i^{(l)} = tanh((W^l * f_i^{l-1}) + b^l) \tag{9}$$

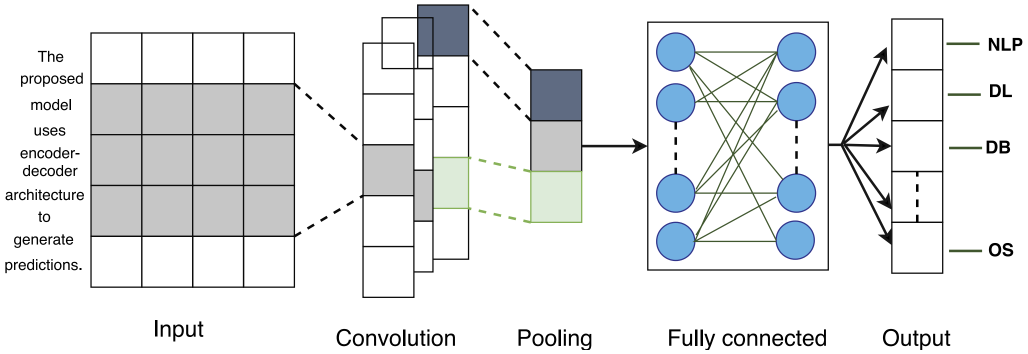

The proposed model uses encoder-decoder architecture to generate predictions.
NLP
DL
DB
OS
Input
Convolution
Pooling
Fully connected
Output

Fig. 7. Convolutional neural network.

where *tanh* is the activation neuron function used. Also, W denotes the parameters of corresponding layer, while f_i^{l-1} represents the segment of layer for the convolution at location i. Additionally, the max-pooling layer down-samples text or image using a sliding window as:

$$f_i^{(l)} = max\left\{f_t^{(l-1)}(i), f_t^{(l-1)}(i+1)\right\}, \tag{10}$$

Using this approach, the computational cost is reduced significantly as the layer shrinks in size. The network performs such operations again and again on different layers, where each layer learns to identify useful features. After learning these features, the network of a CNN behaves like a classifier. The next to last layer computes the probabilities for each class of any item/paper being classified. Finally, the last layer of the network works as a classification layer and produces classification output.

In recent years, CNNs models have gained success in the field of NLP and recommender systems. Compared to multi-layer perceptrons, CNN reduces the number of neurons through pooling. In addition, the shared-weights architecture reduces the number of the parameters. This leads to lower complexity and faster adaptation. To this end, CNNs have shown promising results in capturing the semantic representations and contextual information of citations in citation recommendation problems. In particular, Yin et al. (2017) proposed personalized convolutional neural network (p-CNN), which generates citation recommendation. That is, it learns representative features concerning an author and exploits them in computing the relevance score between the context and the corresponding referred articles. It employs a discriminative training strategy to learn the parameters by minimizing the following:

$$Loss(\theta) = max\left\{0, 1 + s(cc, D_j^-) - s(cc, D^+)\right\} \tag{11}$$

where $s(cc, D)$ denotes the similarity score between context cc and document D, which is computed as the output of multi-layer perceptions in fully connected layer. In this connection, POLAR (Du et al., 2019) proposed an attention-based CNN model to produce citation recommendations. The model exploits the attention matrix for text similarity, in which the importance of a term is calculated using the local and global weights. Finally, the attention matrix and matching matrix are given as an input to the CNN network to exploit different levels of textual similarities.

On different direction, Jeong et al. (2019) proposed a context-aware model simply by using BERT (Devlin et al., 2018) model and a variation of a Graph Convolution Network (GCN) (Kipf & Welling, 2016). To generate textual embedding, it utilizes pre-trained BERT as a context encoder, while the GCN model is used for generating graph embedding. This way, the model exploits both textual content and citation network to produce context-aware citation recommendations.

Recurrent Neural Network (RNN): are similar to the CNNs, however, they preserve information learned, and apply it to upcoming inputs. In RNNs, nodes connections establish a temporal sequential directed graph. Traditional neural networks are not capable of persisting previous information for the upcoming decisions and events, and it is one of their shortcomings. Recurrent neural networks tackle this issue and persist the information by using memory as depicted in Fig. 8 (Mu, 2018). In the diagram, A denotes a layer of feed-forward neural network, while x_t represents an input and the model outputs a value h_t. Here, loops are employed in passing information from one step to the next in the network. Actually, the RNN architecture has many copies of the same graph, where each participating network passes a message to its successor.

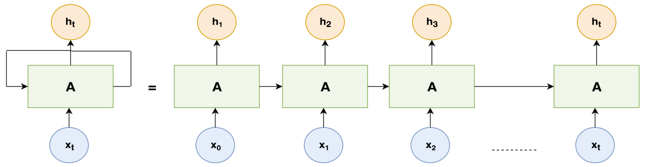


Fig. 8. A simple architecture of RNN.

The ability to remember and identify the patterns encountered across time periods, RNNs are extremely efficient and suitable when applied to data that are sequential. RNNs contain an input, an output and hidden units that play a crucial role in storing information. In contrast to MLPs, RNNs have introduced a directional loop that stores the previous information and apply it to the output. Additionally, RNNs use (1) a dynamic user u_{ut} and paper v_{pt} learned from the long-short-term memory (LSTM), along with (2) the respective static attributes u_u and v_p which are learned using Matrix factorization. In this connection, predictions for a paper can be computed as follows:

$$\hat{r}_{up|t} = f(u_{ut}, v_{pt}, u_u, v_p) \tag{12}$$

To generate accurate predictions, it attempts to minimize the error by bringing predicted score closer to the actual score. Many variants of RNNs are available, which are used depending on the application requirements, however, the most commonly used methods are long short-term memory (LSTM) (Goodfellow et al., 2016), and gated recurrent unit (GRU) (Goodfellow et al., 2016). LSTM is a time recurrent neural network which is suitable to predict important events that have relatively long intervals.

Traditional RNNs can encounter the short-term memory problem when they deal with long sequences. For instance, if we want to process a lengthy textual paragraph to generate predictions, traditional RNN's may miss the necessary information. Additionally, RNNs encounter the vanishing gradient problems, where gradients shrink as it back-propagates through time. Thus, such small values do not contribute too much in learning. Therefore, those layers that face this problem stops the learning process. As these layers do not participate in learning, consequently RNN's encounter short-term memory problem. To overcome this problem, LSTM integrates both short and long-term memories

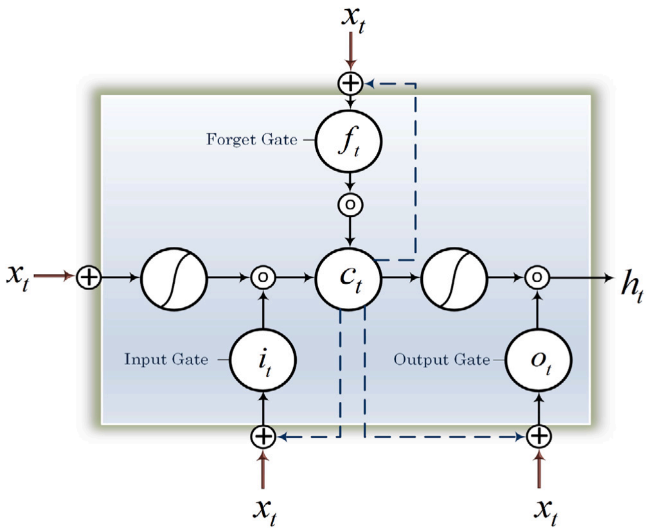


Fig. 9. Long short term memory block.

through a subtle gate control. The difference between these two is that LSTM adds a ‘cell’ to justify whether the information it possesses is utile or not. For instance, a cell memory contains three cells such as an input, a forget and an output gate as depicted in Fig. 9. Mathematically they are represented in Eq. (13) (Abro et al., 2019):

$$\begin{aligned} i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i) \\ f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f) \\ o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o) \\ \widetilde{c_t} &= tanh(W_c x_t + U_c h_{t-1} + b_c) \\ c_t &= f_t \odot c_{t-1} + i_t \odot \widetilde{c_t} \\ h_t &= o_t \odot tanh(c_t), \end{aligned} \tag{13}$$

where σ and $\odot$ denotes the sigmoid function and the element-wise multiplication, respectively. Also, t represents time step, i_t denotes an input gate, forget gate is represented by f_t, On the other hand, o_t is used as an output, c_t denotes a cell state, and h_t represents a hidden one. While W, U and b represent the parameters of the LSTM. The cell state transfers relative information to the next level, and only useful information is kept during training, otherwise forgotten using the forgot gate.

In contrast, the GRUs instead of using the cell state, pass information by employing the hidden state. Additionally, two new gates have been introduced namely the reset, and one update gate. The work of the later one is similar to the former of an LSTM network. While the reset gate decides which past information to ignore. In GRUs, few number of tensor operations are performed compared to LSTMs, therefore they are faster comparatively.

To this end, CACR (Yang et al., 2018) employed LSTM to learn the distributed representation of articles context, and manuscripts. Then, the model selects the top candidate articles based on the relevance scores between these two. On that direction, a CF-based model ML-DTR (Bansal et al., 2016) generates the latent representation of text sequence employing gated recurrent units (GRUs) to produce paper recommendations. In this connection, another RNN-based model (Hassan, 2017) that discovers the latent semantic features of scientific papers and generates personalized recommendations consider users’ feedbacks. More specifically, the study examines the impact of employing word2vec and LSTM to extract the semantic representation of the content of papers.

Generative Adversarial Network (GAN): GANs are comprised of two neural networks namely a Discriminator and a Generator (Goodfellow et al., 2014). The Generator produces fake random number of data of images, text, etc. and tries to fool the later one, while it attempts to identify the fake and real samples. During training, both neural models run in competition with each other. In order to get succeeded in their respective jobs, both the models repeat the corresponding steps many times. The GANs have been designed as a min–max game. In particular, they aim to minimize the reward of Discriminator to maximizing the probability of the Discriminator to make an error. While the Discriminator attempts to minimize its loss, and estimates the probability by considering that the sample it received is from the probe set and not the generated one as shown in Fig. 10.

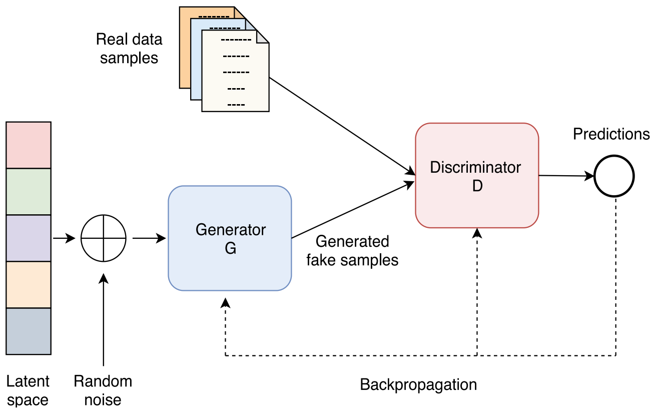


Fig. 10. Framework of generative adversarial network.

Formally, suppose if $p_{true}(d \mid q_n, r)$ represents the user’s relevance distribution then the Generator $p_\theta(d \mid q_n, r)$ attempts to approximate the actual relevance distribution. In contrast, the Discriminator $f_\phi(q \mid d)$ distinguishes between the relevant items/papers and non-relevant items. The main objective function is given as:

$$\begin{aligned} J^{G^*,D^*} &= min_\theta max_\phi \sum_{n=1}^{N} \left(E_{d\sim p_{true}(d|q_n,r)}[log D(d \mid q_n)] \right) \\ &\quad + \left(E_{z\sim p_\theta(d|q_n,r)}[1 - log D((d \mid q_n))] \right) \end{aligned} \tag{14}$$

Here $D(d \mid q_n) = \sigma(f_\phi(q \mid d))$, where σ denotes the sigmoid. Whereas θ is the parameter for generative retrieval while ϕ is employed for the discriminative retrieval. Gradient descent is employed for learning both parameters.

To this end, VCGAN (Zhang et al., 2018) which employs Generative Adversarial Network to generate personalized citation recommendations. Specifically, it exploits the network and the content related to nodes to generate content-based graph representations. Additionally, the model explore the network of the co-authorship. Then, the aforementioned representations are concatenated to get the node feature vectors. Finally, these vectors are employed in generating citation recommendations against a query manuscript as follows:

$$s(q, p_j) = \frac{q \cdot p_j}{\|q\| \cdot \left\|p_j\right\|} \tag{15}$$

where q and p_j represents query and candidate papers, respectively. Similarly, another similar learning method for heterogeneous bibliographic network is GAN-HBNR (Cai, Han & Yang, 2018). The model generates personalized recommendations using the network structure along with the authors, the papers, and the query manuscript to learn representations. The content representation of each vertex is obtained using doc2vec (Le & Mikolov, 2014) embedding approach. To exploit both the network structure and contents of vertices, the model employs Denoising Autoencoder (DAE) network. That is, it uses a feed-forward generator network $G(z)$ that takes a vector $z \in \Re^{h_g}$ as an input and produces a generated vector. While the discriminator network $D(x)$

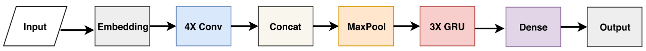


Fig. 11. Architecture of CNRN model.

takes vectors $x \in \Re^{m+n}$ and produces an energy estimate $E \in \Re$. Once the network representations are learned, the model use the similarity computation method between the vectors to find top ranked papers as:

$$\vec{r}_q = \vec{V}_{PR}\vec{V}_{qt}^T + \vec{V}_{AR}\vec{V}_{qa}^T \tag{16}$$

where $\vec{V}_{PR}, \vec{V}_{qt}, and \vec{V}_{qa}$ denote the vector representations of training papers, manuscript text and manuscript author, respectively. Finally, MAAE (Galke et al., 2018) integrates generative adversarial networks with auto-encoders to generate citation as well as label recommendations.

Deep Reinforcement Learning (DRL): is a ML approach where an agent takes an action at a time step t within an environment. In return, the environment returns two kinds of responses to the agent (1) a reward that provides quantitative feedback on the action that the agent took at a particular timeset t, and (2) a state where the environment changes in response to an agent's action. These steps are repeated until it reaches some terminal state. In DRL, reinforcement learning (RL) and deep learning models are combined to solve more complex NLP and IR problems. DRL is employed in citation recommendation models to rank future candidate citations scores for a target user. It employs gradient decent operation to approximate values considering the preference dynamics of a researcher. To this end, TMR-PCR (Cai, Han & Li, 2018) integrates researchers, venues and papers into a three-layered graph through a mutually reinforcing manner to generate citation recommendations. To generate personalized recommendations, the model exploits the researcher's information along with query textual information into a random walk process. It employs a three-layered interactive clustering method over the nodes and mitigates the computational complexity problem linked with random walk methods when applied to a large size graphs. Similarly, MMRQ (Mu et al., 2018) exploits a multi-layered graph (consists of authors, papers, and keywords entities) and employs multi-layer reinforcement rules in the graph to generate query-oriented citation recommendations.

Hybrid: to further improve the performance of citation recommendations, different models have combined two or more than two deep learning networks (i.e., CNN and RNN, CNN and AE, GAN and AE, etc.) to generate citation recommendations. In this survey, we discovered that few models employed hybrid deep learning methods for producing citation recommendations as depicted in Table 2 under column Methodologies and models. In this direction, Ebesu and Fang (2017) adopts an encoder–decoder (ED) approach to learn the semantic relations between citation contexts and relevant cited papers by using author's relations. First, the encoder encodes a given citation context into vector space using a variant of the CNN called time delay neural network. Finally, the GRU decoder exploits attention mechanism and author networks to decode the encoder's representation. (See Fig. 11).

Likewise, attention-based encoder–decoder (AED) (Yang, Zhang, Cai & Dai, 2019) model generates context-aware citation recommendations by utilizing TDNN and RNN as an encoder and decoder networks, respectively. To learn semantic relations between citation context and research papers, the model adopts attention mechanism using the author and venue information.

2.4. Recommendation types

Here we discuss the explored models considering the recommendation types they provide. These models generate three kinds of recommendations namely: (1) local citation recommendation, (2) global citation recommendations, and (3) tags/labels, as depicted in Table 2. Local citation recommendation is when the model generates citation recommendations for a specific context of a paper where a citation is needed, these models are also called context-aware citation recommendation. In contrast, global citation recommendation aims to produce recommendations for a query manuscript. Additionally, some papers generate tag/label recommendations along with local or global recommendations. It is clear that the majority of the models, i.e., 23 out of 35 provide global citation recommendation services, while local/context-aware recommendations stand the second spot as adopted in the 12 citation recommendation models. Also, only one model (Galke et al., 2018) supports label recommendations which reveal that very few studies have targeted label recommendations.

2.5. Problems faced

This section discusses the two main problems researchers faced in conducting their research that include: 1) cold-start, and 2) sparsity. Cold-start papers (Ali et al., 2020) and cold-start users (Christoforidis et al., 2018a; Rafailidis et al., 2017) are the two notorious problems linked with recommendation models. In cold-start case, the model cannot produce quality recommendations due to the unavailability of useful information. It arises when a newly entered paper or user does not have the necessary information using which system can produce adequate predictions (Ali et al., 2020; Kefalas et al., 2018). To overcome such problems, two models (Bansal et al., 2016; Ma & Wang, 2019) have proposed different solutions. In contrast, sparsity is a very common problem causes due to lack of users' ratings that results in low accuracy (Ali et al., 2020). Sparsity problem can be more critical in paper recommendation models as it leads in non-personalized recommendations. It is evident from the literature that only two studies (Bansal et al., 2016; Galke et al., 2018) have addressed this problem.

2.6. Personalization

This subsection is dedicated to the recommendation types given to a target user that are: personalized, non-personalized, and group-based as depicted in Table 2. Personalized recommendation models exploit users' profile information and history to suggest recommendations. In contrast, non-personalized models generate a recommendation list considering popular or top-rated papers. That is, all users receive a similar kind of recommendation results without taking into account the research interests of a single researcher. We can notice in surveyed studies that plenty of the models have produced personalized recommendations.

3. Datasets and metrics for evaluation

To evaluate models performance, researchers have used various datasets and evaluation metrics. As novel recommendation algorithms are emerging, therefore it is requisite to determine an appropriate dataset/s and metric/s that could be used for evaluating the experimental results. Next we provide an overview of the datasets and metrics adopted by the explored models.

3.1. Datasets

In this section, we present an analysis of the most popular datasets depicted in Tables 3. Also, in Table 4 we denote which dataset used from each model so that it will be easier for the readers to choose one depending on the problem they aim to address. The results reveal that, ACL anthology is the most popular dataset used 13 times (Cai, Han & Li, 2018; Cai, Han & Yang, 2018; Cai et al., 2019; Chen et al., 2019a, 2019b; Dai et al., 2019; Jeong et al., 2019; Kobayashi et al., 2018; Mu et al., 2018; Yang, Zhang, Cai & Dai, 2019; Yang, Zhang, Cai & Guo, 2019; Yang et al., 2018; Zhang et al., 2018) as shown in

Table 3
Datasets specifications.

Datasets	Papers	Authors/ users	Venues	Citation relations	Citation context	Tags	Key-phrases	Title	Abstract	Full text	Year info	Ratings	Release year
DBLP[a]	4,107,340	318,406	23,709	✓	–	–	–	✓	✓	–	✓	–	2019
ACL anthology[b]	23766	18862	373	✓	–	–	–	✓	✓	✓	–	–	2016
RefSeer^{x}[c]	855,735	–	–	✓	✓	–	✓	–	✓	✓	✓	–	2015
CiteUlike[d]	5551	16980	–	✓	–	✓	–	✓	✓	–	–	✓	2013
Aminer[e]	3,680,007	212,567	12,770	✓	–	–	–	✓	✓	–	✓	–	2017
PubMed (Bhagavatula et al., 2018)	789	2,12,312	2,319	✓	–	✓	✓	✓	✓	✓	–	–	2015
RARD II[f]	24,00000	–	–	–	–	–	–	–	–	–	–	✓	2018
OpenCorpus[g]	60,90,000	80,30,000	23,672	✓	–	–	✓	✓	✓	–	✓	–	2018
arXiv CS[h]	90,278	269,194	1,489	✓	✓	–	✓	✓	✓	–	✓	–	2018
Scholarly Dataset[h]	100351	50	–	✓	–	–	–	–	✓	✓	–	–	2013

Table 4. It is because the dataset provides access to richer information about the papers, the authors, the venues, the citation relations, and the content. Additionally, RefSeer^{x}, and arXiv CS datasets provide information including the contents of papers and citation contexts, therefore these datasets are suitable for context-aware citation recommendations (Ebesu & Fang, 2017; Huang et al., 2015). Moreover, most recent models (Cai et al., 2019; Ganguly & Pudi, 2017; Mu et al., 2018; Yang, Zhang, Cai & Guo, 2019; Yang et al., 2018) exploiting network embedding techniques use both DBLP and ACL Anthology since they contain rich content information including. Also, a great number of articles use self-collected datasets from different online repositories and libraries such as Springer, ScienceDirect, ACM, and IEEE, etc. The possible reason can be that existing datasets do not fulfill the requirements of their proposed method. Finally, we notice that two of the datasets i.e., CiteUlike and RARD II provide user ratings information, therefore these datasets can easily be adopted for CF-based models (Bansal et al., 2016).

3.2. Evaluation metrics

In this section, we present the most popular evaluation metrics used in the literature that are Precision, Recall, F-Measure, Root Mean Square Error (RMSE), Coverage, normalized Discounted Cumulative Gain (nDCG), and Accuracy. In Table 5, we can see that the majority of models use more than one metric. In particular, six models (Bulut et al., 2020; Dai et al., 2019; Ebesu & Fang, 2017; Färber et al., 2018a, 2018b; Kong et al., 2019) perform a thorough evaluation using 4 metrics, eleven models (Cai, Han & Li, 2018; Cai, Han & Yang, 2018; Chen et al., 2019b; Huang et al., 2015; Jeong et al., 2019; Jiang et al., 2018; Ma & Wang, 2019; Yang, Zhang, Cai & Dai, 2019; Yang, Zhang, Cai & Guo, 2019; Yang et al., 2018; Zhang & Ma, 2020) use 3 metrics, while eight models (Bhagavatula et al., 2018; Cai et al., 2019; Chen et al., 2019a; Gupta & Varma, 2017; Hassan, 2017; Mu et al., 2018; Yin et al., 2017; Zhang et al., 2018) use 2. On the contrary, the remaining nine models (Bansal et al., 2016; Du et al., 2019; Galke et al., 2018; Khadka & Knoth, 2018; Kobayashi et al., 2018; Li et al., 2019; Ma et al., 2019; Sharma et al., 2017; Tian & Zhuo, 2017) use just one metric. It is notable that the more metrics used, the thorough analysis of the model is. Also, the higher the number of metrics the better our understanding of models impact is.

Table 4
Evaluation datasets.

	Models	CiteUlike	CiteSeer	DBLP	ACL Anthology	Aminar	Self collected	arXiv CS	Scholarly	PubMed	OpenCorpus	RARD
1	CIRec (Chen et al., 2019b)	–	–	✓	✓	–	–	–	–	–	–	–
2	RBM-CS (Tang & Zhang, 2009)	–	✓	–	–	–	–	–	–	–	–	–
3	MMRQ (Mu et al., 2018)	–	–	–	✓	–	–	–	–	–	–	–
4	MAAE (Galke et al., 2018)	–	–	–	–	–	–	–	–	✓	–	–
5	HGRec (Ma & Wang, 2019)	–	–	✓	–	–	–	–	–	–	–	–
6	POLAR (Du et al., 2019)	–	–	–	–	✓	–	–	–	–	–	✓
7	RI-PR (Bulut et al., 2020)	–	–	–	–	–	✓	–	–	–	–	–
8	DocCit2Vec (Zhang & Ma, 2020)	–	–	✓	✓	–	–	–	–	–	–	–
9	DRDF-CR (Kobayashi et al., 2018)	–	–	–	✓	–	–	–	–	–	–	–
10	HRLHG (Jiang et al., 2018)	–	–	–	–	–	✓	–	–	–	–	–
11	ML-DTR (Bansal et al., 2016)	✓	–	–	–	–	–	–	–	–	–	–
12	SAR (Gupta & Varma, 2017)	–	–	–	–	✓	–	–	–	–	–	–
13	BERT-GCN (Jeong et al., 2019)	–	–	–	✓	–	✓	–	–	–	–	–
14	NCN (Ebesu & Fang, 2017)	–	✓	–	–	–	–	–	–	–	–	–
15	PPR-DL (Hassan, 2017)	–	–	–	–	–	–	–	–	✓	–	–
16	ASL (Dai et al., 2019)	–	✓	✓	✓	–	–	–	–	–	–	–
17	CITEWERTs(a) (Färber et al., 2018b)	–	–	–	–	–	–	✓	✓	–	–	–
18	CITEWERTs(b) (Färber et al., 2018a)	–	–	–	–	–	–	✓	✓	–	–	–
19	WHIN-CSL (Chen et al., 2019a)	–	–	✓	✓	–	–		–	–	–	–
20	NNRank (Bhagavatula et al., 2018)	–	–	✓	–	–	–	–	–	✓	✓	–
21	GAN-HBNR (Cai, Han & Yang, 2018)	–	–	✓	✓	–	–	–	–	–	–	–
22	PCCR (Yang et al., 2018)	–	–	✓	✓	–	–	–	–	–	–	–
23	NPM (Huang et al., 2015)	–	✓	–	–	–	–	–	–	–	–	–
24	Paper2vec (Tian & Zhuo, 2017)	–	–	–	–	–	✓	–	–	–	–	–
25	VOPRec (Kong et al., 2019)	–	–	–	–	–	✓	–	–	–	–	–
26	p-CNN (Yin et al., 2017)	–	–	–	–	–	✓	–	–	–	–	–
27	VCGAN (Zhang et al., 2018)	–	–	–	✓	–	–	–	–	–	–	–
28	TMR-PCR (Cai, Han & Li, 2018)	–	✓	✓	✓	–	–	–	–	–	–	–
29	CPR (Sharma et al., 2017)	–	–	–	–	–	✓	–	–	–	–	–
30	BNR (Cai et al., 2019)	–	–	✓	✓	–	–	–	–	–	–	–
31	HRM (Li et al., 2019)	–	–	–	–	–	✓	–	–	–	–	–
32	CPA-CE (Khadka & Knoth, 2018)	–	–	–	–	–	✓	–	–	–	–	–
33	AED (Yang, Zhang, Cai & Dai, 2019)	–	–	✓	✓	–	–	–	–	–	–	–
34	NREP (Yang, Zhang, Cai & Guo, 2019)	–	–	✓	✓	–	–	–	–	–	–	–
35	HIPRec (Ma et al., 2019)	–	–	✓	–	–	–	–	–	–	–	–

4. Comparative analysis of the models

In this section, we analyze and compare the results of explored citation recommendation approaches. In this analysis, we consider only the models which use the same set of evaluation metrics and datasets. Models that use self-collected data are excluded for reasons of a fair comparison.

4.1. Comparison on the DBLP dataset

In Fig. 12(a), we notice that among the models using this dataset PCCR outperforms others in terms of MAP, MRR and recall. This is due to the fact that the model employs LSTM and BiLSTM networks to learn the distributed representations of scientific papers and citation

[a] https://www.aminer.cn/citation.
[b] http://clair.eecs.umich.edu/aan/index.php.
[c] https://psu.app.box.com/v/refseer.
[d] https://github.com/js05212/citeulike-a.
[e] https://www.aminer.cn/data.
[f] http://data.mr-dlib.org.
[g] https://allenai.org/data/data-all.html.
[h] http://www.citation-recommendation.org/publications.

Table 5
Evaluation metrics.

	Models	Precision	Recall	RMSE	Coverage	nDCG	MRR	F-Measure	Accuracy	Novelty
1	CIRec (Chen et al., 2019b)	✓	✓	–	–	✓	–	–	–	–
2	RBM-CS (Tang & Zhang, 2009)	✓	–	–	–	–	✓	–	–	–
3	MMRQ (Mu et al., 2018)	–	✓	–	–	✓	–	–	–	–
4	MAAE (Galke et al., 2018)	–	–	–	–	–	✓	–	–	–
5	HGRec (Ma & Wang, 2019)	✓	✓	–	–	–	–	✓	–	–
6	POLAR (Du et al., 2019)	–	–	–	–	✓	–	–	–	–
7	RI-PR (Bulut et al., 2020)	✓	✓	–	–	–	–	✓	✓	–
8	DocCit2Vec (Zhang & Ma, 2020)	✓	✓	–	–	✓	–	–	–	–
9	DRDF-CR (Kobayashi et al., 2018)	–	–	–	–	✓	–	–	–	–
10	HRLHG (Jiang et al., 2018)	✓	–	–	–	✓	✓	–	–	–
11	ML-DTR (Bansal et al., 2016)	–	✓	–	–	–	–	–	–	–
12	SAR (Gupta & Varma, 2017)	–	✓	–	–	✓	–	–	–	–
13	BERT-GCN (Jeong et al., 2019)	✓	✓	–	–	–	✓	–	–	–
14	NCN (Ebesu & Fang, 2017)	✓	✓	–	–	✓	✓	–	–	–
23	PPR-DL (Hassan, 2017)	–	✓	–	–	✓	–	–	–	–
16	ASL (Dai et al., 2019)	✓	✓	–	–	✓	–	–	–	–
17	CITEWERTs(a) (Färber et al., 2018b)	✓	✓	–	–	–	–	✓	✓	–
18	CITEWERTs(b) (Färber et al., 2018a)	✓	✓	–	–	–	–	✓	✓	–
19	WHIN-CSL (Chen et al., 2019a)	–	✓	–	–	✓	–	–	–	–
20	NNRank (Bhagavatula et al., 2018)	–	✓	–	–	✓	–	–	–	–
21	GAN-HBNR (Cai, Han & Yang, 2018)	✓	✓	–	–	–	✓	–	–	–
22	PCCR (Yang et al., 2018)	✓	✓	–	–	–	✓	–	–	–
23	NPM (Huang et al., 2015)	✓	–	–	–	✓	✓	–	–	–
24	Paper2vec (Tian & Zhuo, 2017)	–	–	–	–	–	–	–	–	✓
25	VOPRec (Kong et al., 2019)	✓	✓	✓	–	✓	–	–	–	–
26	p-CNN (Yin et al., 2017)	✓	✓	–	–	–	–	–	–	–
27	VCGAN (Zhang et al., 2018)	–	✓	–	–	✓	–	–	–	–
28	TMR-PCR (Cai, Han & Li, 2018)	✓	✓	–	–	–	✓	–	–	–
29	CPR (Sharma et al., 2017)	–	–	–	–	✓	–	–	–	–
30	BNR (Cai et al., 2019)	✓	–	–	–	–	✓	–	–	–
31	HRM (Li et al., 2019)	✓	–	–	–	–	–	–	–	–
32	CPA-CE (Khadka & Knoth, 2018)	–	–	–	–	✓	–	–	–	–
33	AED (Yang, Zhang, Cai & Dai, 2019)	✓	✓	–	–	✓	–	–	–	–
34	NREP (Yang, Zhang, Cai & Guo, 2019)	✓	✓	–	–	✓	–	–	–	–
35	HIPRec (Ma et al., 2019)	–	–	–	–	–	–	–	✓	–

contexts, respectively. The significance of PCCR lies in its capability of exploiting the semantic representations of research papers with additional sources such as authors, venues, and textual content to generate quality citation recommendations. The second model in that ranking is GAN-HBNR, which exploits the network structure along with the content related to authors, papers, and query paper to generate latent representations of these vertices. The content representation of each vertex is obtained using the doc2vec embedding approach. Then the model learns the network structure and content information using the energy-based generative adversarial network.

4.2. Comparison on the CiteSeer dataset

Similarly, in Fig. 12(b) we notice that BNR gained the highest performance in terms of MRR and recall. This model explores the structure of bibliographic network and relevant textual content correspond to papers, venues and authors to learn their latent representations over the paths that are created through random walk. During training the Skip-Gram, Node2vec is employed on the corpus generated to obtain the low-dimensional representations of network objects. Finally, these embeddings are used to produce relevant citation recommendations. On the other hand, ASL (Dai et al., 2019), comes second since it extends the basic SDAE into an attentive SDAE (ASDAE), which enables the model to capture the global attention from referred context when encoding an article. To generate the vector representation of referred context and article, the model extends Bidirectional LSTM (Bi-LSTM) by introducing a local attention-layer. Additionally, the model utilizes the author's information to generate the vector representations of cited paper and citation context.

4.3. Comparison on the ACL anthology dataset

Finally, we can observe in Fig. 12(c) that Bert-GCN performed higher in terms of MRR and recall compared to its counterparts since the model employs BERT and Graph Convolution Networks (GCN) to produce context-aware recommendations. The model generates textual embedding by utilizing pre-trained BERT as a context encoder, while GCN model is employed for network representation. In a nutshell, those models that capture the semantic representations of papers and utilize auxiliary sources such as contents, authors profile, and venue information can better exploit researcher's preferences and produce more robust results.

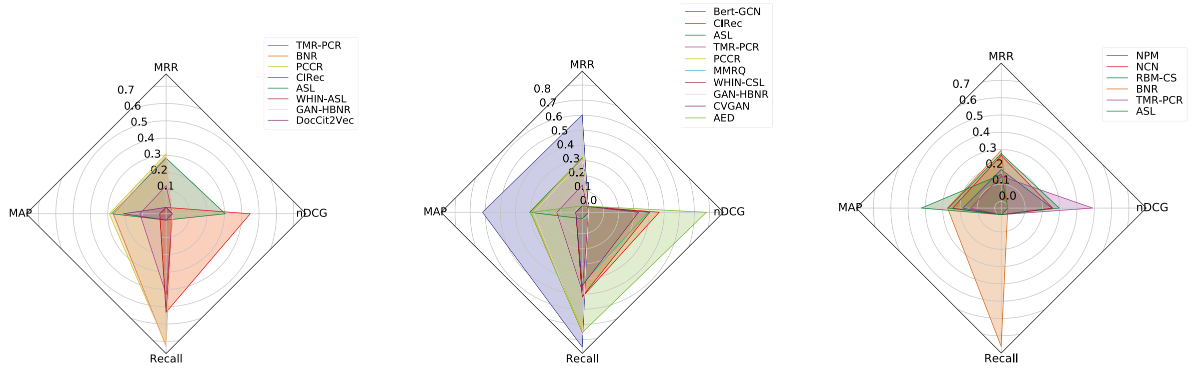
4.4. Performance over metrics

Regarding the performance of the models over the metrics examined, next, we present some of our findings. The first metric we analyze is MRR, which examines how relevant is the first item ranked for a target user. To this point, we notice that PCCR, BNR, and Bert-GCN outperform other models for the DBLP, CiteSeer, and ACL datasets, respectively. The reason beneath is that PCCR uses the LSTM model to learn the embeddings of scientific papers combining its content, author, and venue information. This way, they manage to learn past preferences that enhance performance for the first item recommended. Similarly, embeddings are also used in the BNR model. This time the model learns inter-node and intra-node relations exploited by random walk algorithm over the given HIN. Finally, they capture node-content correlations by maximizing the co-occurrence of word sequence given a node. In the same direction, Bert-GCN employs BERT as a context encoder to extract textual content and Graph Convolution Network as a citation encoder to learn the embeddings. Considering all the aforementioned models to find the most suitable item to recommend, we conclude that embeddings are the most popular approach. Unfortunately, this metric emphasizes just on the first item and ignores the rest of the items in the list, thus it is not reliable to overall performance.

The second metric computes the mean average precision (MAP) of all the list through an item and represents the area under the Precision–Recall curve. Also, it gives more weight to errors that happen high up in the top@n list. We notice that PCCR, RBM-CS, and Bert-GCN generated significant results over the MAP metric for the DBLP, CiteSeer, and ACL datasets, respectively. Considering this metric, we have two main approaches. The first one uses embeddings to learn information, while the second one employs the Restricted Boltzmann Machine. In particular, the model aims to learn a mixture of topic distribution over paper contents and citation relationships. The drawback of MAP is that it does not consider recommendations as an ordered list and does not fit for fine-grained numerical ratings. Thus, the position of the ranked item is not that important which does not fit for evaluations that consider top@n rankings.

The third metric is Recall that measures the fraction of the relevant items retrieved successfully. Close to MRR, once again, PCCR, BNR, and Bert-GCN have a clear edge over other models in terms of this metric as happens in the first metric presented. To this point, we notice that the use of embeddings is not only reliable while trying to recommend the best item on the top@n list for a target user, but also it is used while aiming to retrieve all the relevant items retrieved successfully on that list. The biggest drawback for this metric is the scenario where no relevant items retrieved in the list which, is quite common due to the sparsity of the data.

The final metric is normalized Discounted Cumulative Gain (DCG), which assesses the significance of items in the list against less important ones. Thus, the position of the recommended items determines how useful they are. Regarding this metric, CIRec, ASL, and AED outperformed other models over the datasets examined. Close to previous methods, the later one uses embeddings to learn the network structure. In particular, CIRec constructs a weighted heterogeneous network of



(a) DBLP (b) ACL Anthology (c) CiteSeer

Fig. 12. Radar analysis against metrics for models using (a) DBLP, (b) ACL Anthology, and (c) CiteSeer datasets.

five matrices to represent the probability of entity-to-entity migration based on citation tendency. Then, biased random-walk explores articles' characteristics (authors and terms) and citation relations. In the same direction, ASL learns the embedding of research papers and citation contexts by combining the Bi-LSTM with a stacked denoising autoencoder (SDAE). The model utilizes an attention mechanism to exploit the global attention from citation context and using Word2vec to embed them into the model. Then, the model consolidates the author's vector with the vector of the citation context using MLP to learn the final embedding of the corresponding cited paper. Similarly, AED integrates Time-Delay Neural Network (TDNN) to learn the low-dimensional representation of citation contexts and RNN as an encoder–decoder network to learn the semantic relations between citation context and scientific papers. Unfortunately, this metric does not penalize the retrieval of unrelated items on the list, which in some cases is not suitable for the examined problem.

5. Research challenges and opportunities

Next, we present some main problems linked with the domain, including: (1) cold-start (user/paper), (2) data sparsity, (3) complex-network analysis, and (4) network-oriented solution. Additionally, we discuss solutions presented in the explored literature to address the aforementioned problems.

5.1. Cold-start

Cold-start papers and cold-start users are two prominent issues in the domain (Christoforidis et al., 2018a, 2018b). In such cases, a model cannot make accurate predictions as it does not have enough information about the corresponding objects. The problem is caused each time a new paper or researcher enters the 'system'. Obviously, we own very limited information about them, thus personalization is even more difficult. For instance, if a new research paper is submitted to the system, and the system does not have its ratings, metadata, and abstract, then it would be hard for the model to produce justifiable recommendations (Ali et al., 2020; Christoforidis et al., 2018a).

To overcome the cold-start problem, most of the models employed articles' textual content, tags, profile, and venue information as data factors over graph-based and matrix-based data representation methods. It is noticeable that all these models addressed the cold-start problem in the context of global citation recommendation. However, these studies have adopted different DL architectures to address these problems. For example, HRM (Li et al., 2019) employs a two-layer feed-forward neural network to exploit the content and user profile and re-rank the candidate articles. That is, it employs different content-based measures extracted using different papers related features, such as word and author similarity from an embedding space to alleviate the cold-start problem. Similarly, Bansal et al. (2016) generates the vector representations of research papers from their textual content using GRU, a variant of RNN neural network, to address the cold-start problem. On the same direction, HGRec (Ma & Wang, 2019) uses a heterogeneous graph embedding technique to exploit the textual contents and graph structure. In contrast to previous, HIPRec (Ma et al., 2019), a heterogeneous network embedding model extracts the topological features of HINs objects (authors, papers, tags, and venue) based on the meta-paths and meta-graphs. In a nutshell, few models have investigated the cold-start problem, however, these studies reveal that embeddings and MLP are the main methodologies adopted. Also, it is noticeable that they aim to incorporate auxiliary side information related to users' or items to alleviate cold-start problem.

5.2. Data sparsity

Is another well-known problem attracted attention in the literature. It occurs when we have big datasets but limited information about each user. Thus, making recommendations in such scenarios is a difficult task since it is hard to correlate users with similar tastes (Khusro et al., 2016). In contrast to other domains of applications such as movies, and POIs, the number of users in paper recommendation domain is usually less than the research articles, which results in the data sparsity problem (Ali et al., 2020). It is noticeable that most of the models aim to alleviate sparsity with the use of auxiliary information to enrich their knowledge about each user by extending the network of relations with new nodes and edges. The new nodes represent relations between paper contents, tags, user profiles, and ratings. The main difference between them is the used deep learning method for personalizing recommendations. For example, MAAE (Galke et al., 2018) uses a combination of a generative adversarial network with an auto-encoder to mitigate the network sparsity problem. The model exploits information, including ratings, metadata, namely the documents' title, as content-based features. The auto-encoder component reconstructs the sparse item vectors, while the discriminator distinguishes between the generated codes and samples from a selected prior distribution. In this way, it learns node embedding by distinguishing the code from a smooth prior. Similarly, ML-DTR (Bansal et al., 2016) use a variant of RNN (i.e., GRU) to encode the textual content of research papers into a vector space. Also, HRM (Li et al., 2019) exploits the content and user behavior to re-rank the article recommendations. The model uses a joint matrix factorization method to learn a mapping through MLP between papers users browsed and clicks on the recommendations, which results in alleviating the sparsity problem linked with the recommendation click data. Finally, GAN-HBNR (Cai, Han & Yang, 2018) uses the network structure along with the authors, the papers, and the query manuscript to learn objects representations.

5.3. Network analysis

Typically, large size papers networks consist of various types of nodes such as authors/researchers, research articles, tags/labels, publishing venues, and time. These objects establish links and relation with each other such as authorship, citations, share-topic, share-venue, representing relations between objects in the graph. Additionally, some networks contain weights over the relations, which demonstrate the significance of relations between the nodes of the network. To produce semantic-aware citation recommendations, it is indispensable to capture such meaningful relations and semantics in the HIN.

Traditional systems (Sugiyama & Kan, 2013; Xia et al., 2016) that perform network analysis consider merely the explicit relations between papers (i.e., citation, co-citation etc.). Therefore, such models ignore auxiliary side information in the network and therefore fail to produce quality recommendations (Cai et al., 2019). Most of the models that produce recommendations based on such ordinary relations ignore complex relationships and semantics among objects because they only exploit uni-partite relationships, therefore cannot comprehend relationships with indirectly linked nodes. On the contrary, traditional graph-based citation recommendation systems apply random walks on the graphs (Brin & Page, 1998), and therefore old papers get more weight compared to a new published one. Thus, a very small part of the network is considered while ranking vertices.

To overcome such problems, Heterogeneous Information Networks (HINs) embedding methods have been proposed to generate quality results. In contrast to traditional graph-based algorithms, they can efficiently model the real world data, capture semantics between nodes, and produce robust results (Ali et al., 2020). These relations may represent papers–authors, papers–publication time, papers–venues, papers–labels, papers–topics, papers–papers, and contents related to these nodes. In literature HINs based models (Cai, Han & Yang, 2018; Cai et al., 2019; Chen et al., 2019a, 2019b; Galke et al., 2018; Jeong et al., 2019; Jiang et al., 2018; Ma & Wang, 2019; Ma et al., 2019; Mu et al., 2018) , employ papers' content, authors' profiles, and citation networks as data feature. Nevertheless, such models utilize different deep learning methods to exploit relations between the nodes.

For instance, Ma and Wang (2019) exploit the contents and network structure based on network embedding method. Initially, the model uses Doc2vec to create the vector representations of the author and paper by utilizing the relevant contents. Then, it updates jointly the node embeddings by employing two meta-path based first-order and second-order proximities. Similarly, Chen et al. (2019a) proposed a weighted network embedding model that uses to exploit the significance of relations and learn the representations of network objects. In particular, the model exploits four kinds of meaningful relations (semantic linking, citing, writing, and co-author) between authors and papers to generate semantic-aware node embedding. Similarly, other models use embedding methods (Chen et al., 2019a, 2019b; Kong et al., 2019; Ma & Wang, 2019; Ma et al., 2019), generative adversarial networks (Cai, Han & Yang, 2018; Galke et al., 2018), and reinforcement learning (Cai, Han & Li, 2018; Mu et al., 2018) methods to explore relations in HINs. In a nutshell, to capture semantics between HINs nodes and learn context-preserving node representations, it is indispensable to use an appropriate deep learning architecture that can better exploit useful relationships.

5.4. Network-oriented solution

Models exploiting the citation network, either ignore the contents and structure of papers (Introduction, Methodology, Results etc.), or consider very limited metadata such as keywords, title and citations (Chakraborty et al., 2015). Such systems encounter more critical issues when there is sparse information or the absence of true citations in systems that produce recommendations only exploiting the network structure. According to Bai et al. (2019) most models which employ graph-based methods do not use the content of research papers and exploit merely the structure of the citation network to recommend citations. Though, exploiting citation network can improve results, but ignoring the content can cause inadequate predictions (Ali et al., 2020). It is noteworthy that authors tend to cite articles that are close in the methodology and results (Habib & Afzal, 2019). For example, to identify a closely related algorithm or an alternative method for a query paper using only the citation network is not an easy task. Thus, the logical structure and papers content need to be exploited. That is the reason why, recent years many models (Cai, Han & Yang, 2018; Chen et al., 2019a; Du et al., 2019; Färber et al., 2018a; Khadka & Knoth, 2018; Yang, Zhang, Cai & Dai, 2019) use the content of research papers along with network structure to employ semantic relations and network proximity between objects. To conclude, traditional graph-based and random walk methods cannot capture meaningful relations and topological features as they treat recommendation as a link prediction problem. Thus, it is pertinent to explore the entire heterogeneous network and relations between the corresponding objects, which include papers, authors, venues, topics, labels, publishing time, and relevant contents.

6. Conclusion

Nowadays, the huge amount of available research articles on digital libraries makes finding relevant papers a difficult task. To address this issue, many deep learning methods were introduced to assist users. This is the first survey that explores the area of citation recommendation domain using deep neural methods. In this survey, we explored and categorized 35 DL-based models using the following six information criteria: data factors utilized, representation methods of data, methodologies and algorithms adopted, the recommendation services/types, problems encountered, and personalization. Additionally, we discussed the well-known evaluation protocols, datasets, and metrics used in the literature. Furthermore, we analyzed and examined the results of explored models using a criteria set for the comparison. Next, we present a collection of findings based on the explored DL-based citation recommendation models. (1) The majority of the explored models in literature employed paper contents, tags, user profiles, and citation networks as data factors, which reveals that using such information can better represent researchers' interests and help understand their needs. (2) Embeddings are the most popular method adopted in 15 out of 35 models presented in this survey. These methods brought notable improvement in accuracy since they exploit the contextual information, semantics, and auxiliary information correspond to research papers and users. Moreover, systems employing heterogeneous network embedding can adequately capture researchers' preference dynamics. (3) The survey revealed that the variants of RNNs, such as the LSTM, and BiLSTM, stand second since they capture long term dependencies and model contextual information correspond to papers' contexts. (4) Models that used auxiliary information including, papers' content, tags, citation networks, and user profiles as data factors, can offer improved recommendations and alleviate the sparsity and cold-start problems since such techniques exploit rich information and enrich knowledge regarding users' interests.

To all these findings, we notice that there are many unexplored areas for future research. Below we highlight some of the points that will attract interest in the future.

- Specialized citation recommendation models can alleviate the sparsity and cold-start problems and improve accuracy.
- Recent NRL methods can employ meaningful relationships and semantics between the objects of HINs, yet no research work has examined the time dimension, which can help in generating appealing recommendations.
- Also, only few models exploit weighted information networks, where the significance of relations between the nodes is represented. Thus embedding methods over weighted HINs can produce semantic-preserving nodes representations and citation recommendations.

- Finally, novel Deep Learning methods have recently been proposed including one-shot and few-shot learning, Adversarial Training, and Transformers etc. However, the application of such networks in this field is limited at the moment.

In conclusion, we believe that the various research paths explored in this survey will serve as supportive material for the research community working on citation recommendation and a stimulating read for beginners joining this active area.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abro, W. A., Qi, G., Gao, H., Khan, M. A., & Ali, Z. (2019). Multi-turn intent determination for goal-oriented dialogue systems. In *2019 international joint conference on neural networks (IJCNN)* (pp. 1–8).

Ali, Z., Qi, G., Kefalas, P., Abro, W. A., & Ali, B. (2020). A graph-based taxonomy of citation recommendation models. *Artificial Intelligence Review*, 1–44.

Bai, X., Wang, M., Lee, I., Yang, Z., Kong, X., & Xia, F. (2019). Scientific paper recommendation: A survey. *IEEE Access*, *7*, 9324–9339.

Bansal, T., Belanger, D., & McCallum, A. (2016). Ask the GRU: Multi-task learning for deep text recommendations. In *Proceedings of the 10th ACM conference on recommender systems (RecSys)* (pp. 107–114). New York, NY, USA.

Batmaz, Z., Yurekli, A., Bilge, A., & Kaleli, C. (2019). A review on deep learning for recommender systems: challenges and remedies. *Artificial Intelligence Review*, *52*(1), 1–37.

Beel, J., Gipp, B., Langer, S., & Breitinger, C. (2016). Research-paper recommender systems: a literature survey. *International Journal on Digital Libraries*, *17*(4), 305–338.

Bhagavatula, C., Feldman, S., Power, R., & Ammar, W. (2018). Content-based citation recommendation. In *Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: Human language technologies, vol. 1* (pp. 238–251). New Orleans, Louisiana.

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics (TACL)*, *5*, 135–146.

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, *30*(1–7), 107–117.

Bulut, B., Gündoğan, E., Kaya, B., Alhajj, R., & Kaya, M. (2020). User's research interests based paper recommendation system: A deep learning approach. In *Putting social media and networking data in practice for education, planning, prediction and recommendation* (pp. 117–130).

Cai, X., Han, J., Li, W., Zhang, R., Pan, S., & Yang, L. (2018). A three-layered mutually reinforced model for personalized citation recommendation. *Transactions on Neural Networks and Learning Systems*, *29*(12), 6026–6037.

Cai, X., Han, J., & Yang, L. (2018). Generative adversarial network based heterogeneous bibliographic network representation for personalized citation recommendation. In *Thirty-second AAAI conference on artificial intelligence*.

Cai, X., Zheng, Y., Yang, L., Dai, T., & Guo, L. (2019). Bibliographic network representation based personalized citation recommendation. *IEEE Access*, *7*, 457–467.

Chakraborty, T., Modani, N., Narayanam, R., & Nagar, S. (2015). DiSCern: A diversified citation recommendation system for scientific queries. In *31st IEEE international conference on data engineering* (pp. 555–566). Seoul, South Korea.

Chen, J., Liu, Y., Zhao, S., & Zhang, Y. (2019). Citation recommendation based on weighted heterogeneous information network containing semantic linking. In *2019 IEEE international conference on multimedia and expo (ICME)* (pp. 31–36).

Chen, X., Zhao, H.-j., Zhao, S., Chen, J., & Zhang, Y.-p. (2019). Citation recommendation based on citation tendency. *Scientometrics*, *121*(2), 937–956.

Christoforidis, G., Kefalas, P., Papadopoulos, A., & Manolopoulos, Y. (2018). Recommendation of points-of-interest using graph embeddings. In *5th IEEE international conference on data science and advanced analytics (DSAA)* (pp. 31–40).

Christoforidis, G., Kefalas, P., Papadopoulos, A., & Manolopoulos, Y. (2018). Recommendation of points-of-interest using graph embeddings. In *2018 IEEE 5th international conference on data science and advanced analytics (DSAA)* (pp. 31–40).

Cui, P., Wang, X., Pei, J., & Zhu, W. (2019). A survey on network embedding. *Transactions on Knowledge and Data Engineering (TKDE)*, (5), 833–852.

Dai, T., Zhu, L., Wang, Y., & Carley, K. M. (2019). Attentive stacked denoising autoencoder with bi-LSTM for personalized context-aware citation recommendation. *Transactions on Audio, Speech, and Language Processing (TACL)*, 1–15.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805.

Du, Z., Tang, J., & Ding, Y. (2019). POLAR: Attention-based CNN for one-shot personalized article recommendation. In *Machine learning and knowledge discovery in databases* (pp. 675–690).

Ebesu, T., & Fang, Y. (2017). Neural citation network for context-aware citation recommendation. In *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval* (pp. 1093–1096).

Färber, M., Thiemann, A., & Jatowt, A. (2018). CITEWERTs: a system combining cite-worthiness with citation recommendation. In *European conference on information retrieval* (pp. 815–819).

Färber, M., Thiemann, A., & Jatowt, A. (2018). To cite, or not to cite? Detecting citation contexts in text. In *European conference on information retrieval* (pp. 598–603).

Galke, L., Mai, F., Vagliano, I., & Scherp, A. (2018). Multi-modal adversarial autoencoders for recommendations of citations and subject labels. In *Proceedings of the 26th conference on user modeling, adaptation and personalization* (pp. 197–205).

Ganguly, S., & Pudi, V. (2017). Paper2vec: Combining graph and text information for scientific paper representation. In *Advances in information retrieval* (pp. 383–395).

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672–2680).

Grover, A., & Leskovec, J. (2016). Node2Vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (KDD)* (pp. 855–864).

Gupta, S., & Varma, V. (2017). Scientific article recommendation by using distributed representations of text and graph. In *Proceedings of the 26th international conference on world wide web companion* (pp. 1267–1268). Republic and Canton of Geneva, Switzerland.

Habib, R., & Afzal, M. T. (2019). Sections-based bibliographic coupling for research paper recommendation. *Scientometrics*, *119*(2), 643–656.

Hassan, H. A. (2017). Personalized research paper recommendation using deep learning. In *Proceedings of the 25th conference on user modeling, adaptation and personalization* (pp. 327–330).

Huang, W., Wu, Z., Liang, C., Mitra, P., & Giles, C. L. (2015). A neural probabilistic model for context based citation recommendation. In *Proceedings of the twenty-ninth conference on artificial intelligence (AAAI)* (pp. 2404–2410).

Jeong, C., Jang, S., Shin, H., Park, E., & Choi, S. (2019). A context-aware citation recommendation model with BERT and graph convolutional networks. arXiv:1903.06464.

Jiang, Z., Yin, Y., Gao, L., Lu, Y., & Liu, X. (2018). Cross-language citation recommendation via hierarchical representation learning on heterogeneous graph. In *The 41st international ACM SIGIR conference on research &#38; development in information retrieval* (pp. 635–644).

Jordan, M. I., Sejnowski, T. J., & Poggio, T. A. (2001). Learning and relearning in Boltzmann machines. In *Graphical models: Foundations of neural computation* (pp. 45–76).

Kefalas, P., Symeonidis, P., & Manolopoulos, Y. (2015). A graph-based taxonomy of recommendation algorithms and systems in LBSNs. *Transactions on Knowledge and Data Engineering (TKDE)*, *28*(3), 604–622.

Kefalas, P., Symeonidis, P., & Manolopoulos, Y. (2018). Recommendations based on a heterogeneous spatio-temporal social network. *World Wide Web*, *21*(2), 345–371.

Khadka, A., & Knoth, P. (2018). Using citation-context to reduce topic drifting on pure citation-based recommendation. In *Proceedings of the 12th ACM conference on recommender systems* (pp. 362–366).

Khusro, S., Ali, Z., & Ullah, I. (2016). Recommender systems: issues, challenges, and research opportunities. In *Information Science and Applications (ICISA) 2016* (pp. 1179–1189).

Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv:1609.02907.

Kobayashi, Y., Shimbo, M., & Matsumoto, Y. (2018). Citation recommendation using distributed representation of discourse facets in scientific articles. In *Proceedings of the 18th ACM/IEEE on joint conference on digital libraries* (pp. 243–251). New York, NY, USA.

Kong, X., Mao, M., Wang, W., Liu, J., & Xu, B. (2019). VOPRec: Vector representation learning of papers with text information and structural identity for recommendation. *Transactions on Emerging Topics in Computing*, 1–12.

Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. In *Proceedings of the 31st international conference on machine learning, vol. 32* (pp. 1188–1196).

Li, X., Chen, Y., Pettit, B., & Rijke, M. D. (2019). Personalised reranking of paper recommendations using paper content and user behavior. *ACM Transactions on Information Systems (TOIS)*, *37*(3).

Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., & Alsaadi, F. E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing*, *234*, 11–26.

Ma, X., & Wang, R. (2019). Personalized scientific paper recommendation based on heterogeneous graph representation. *IEEE Access*, *7*, 79887–79894.

Ma, S., Zhang, C., & Liu, X. (2020). A review of citation recommendation: from textual content to enriched context. *Scientometrics*, 1–28.

Ma, X., Zhang, Y., & Zeng, J. (2019). Newly published scientific papers recommendation in heterogeneous information networks. *Mobile Networks and Applications*, *24*(1), 69–79.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th international conference on neural information processing systems, vol. 2* (pp. 3111–3119). Red Hook, NY, USA.

Mu, R. (2018). A survey of recommender systems based on deep learning. *IEEE Access*, *6*, 69009–69022.

Mu, D., Guo, L., Cai, X., & Hao, F. (2018). Query-focused personalized citation recommendation with mutually reinforced ranking. *IEEE Access*, *6*, 3107–3119.

Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). DeepWalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 701–710).

Rafailidis, D., Kefalas, P., & Manolopoulos, Y. (2017). Preference dynamics with multimodal user-item interactions in social media recommendation. *Expert Systems with Applications*, *74*, 11–18.

Sharma, R., Gopalani, D., & Meena, Y. (2017). Concept-based approach for research paper recommendation. In *Pattern recognition and machine intelligence* (pp. 687–692).

Shi, C., Hu, B., Zhao, W. X., & Yu, P. S. (2019). Heterogeneous information network embedding for recommendation. *Transactions on Knowledge and Data Engineering (TKDE)*, *31*, 357–370.

Son, J., & Kim, S. B. (2017). Academic paper recommender system using multilevel simultaneous citation networks. *Decision Support Systems*, *105*, 24–33.

Sugiyama, K., & Kan, M.-Y. (2013). Exploiting potential citation papers in scholarly paper recommendation. In *Proceedings of the 13th joint conference on digital libraries (JCDL)* (pp. 153–162).

Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., & Mei, Q. (2015). LINE: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web (WWW)* (pp. 1067–1077).

Tang, J., & Zhang, J. (2009). A discriminative approach to topic-based citation recommendation. In *Pacific-Asia conference on knowledge discovery and data mining (KDD)* (pp. 572–579).

Tian, G., & Jing, L. (2013). Recommending scientific articles using bi-relational graph-based iterative RWR. In *Proceedings of the 7th ACM conference on recommender systems (RecSys)* (pp. 399–402).

Tian, H., & Zhuo, H. H. (2017). Paper2vec: Citation-context based document distributed representation for scholar recommendation. arXiv:1703.06587.

Xia, F., Liu, H., Lee, I., & Cao, L. (2016). Scientific article recommendation: Exploiting common author relations and historical preferences. *Transactions on Big Data*, *2*(2), 101–112.

Yang, L., Zhang, Z., Cai, X., & Dai, T. (2019). Attention-based personalized encoder-decoder model for local citation recommendation. *Computational Intelligence and Neuroscience*, *2019*.

Yang, L., Zhang, Z., Cai, X., & Guo, L. (2019). Citation recommendation as edge prediction in heterogeneous bibliographic network: A network representation approach. *IEEE Access*, *7*, 23232–23239.

Yang, L., Zheng, Y., Cai, X., Dai, H., Mu, D., Guo, L., & Dai, T. (2018). A LSTM based model for personalized context-aware citation recommendation. *IEEE Access*, *6*, 59618–59627.

Yin, J., Li, X., Chen, L., Jensen, C. S., Shahabi, C., Yang, X., & Lian, X. (2017). Personalized citation recommendation via convolutional neural networks. In *Web and Big Data* (pp. 285–293).

Zhang, Y., & Ma, Q. (2020). Citation recommendations considering content and structural context embedding. arXiv:2001.02344.

Zhang, Y., Yang, L., Cai, X., & Dai, H. (2018). A novel personalized citation recommendation approach based on gan. In *International symposium on methodologies for intelligent systems* (pp. 268–278).

Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys*, *52*(1), 5.