Journal of Scientometric Research, 2024; 13(1):217-229.
https://www.jscires.org

Research Article

Keyphrase-Based Literature Recommendation: Enhancing User Queries with Hybrid Co-citation and Co-occurrence Networks

Mayur Makwana, Rupa Mehta

Department of Computer Science and Engineering, Sardar Vallabhbhai National Institute of Technology, Surat, Gujarat, INDIA.

ABSTRACT

The literature recommendation system addresses the issue of time-consuming literature searches for researchers. A scholarly literature recommendation system recommends related papers to the user's search query. Systems can improve the precision of user queries by generating relevant keywords. The proposed approach aimed to recommend research papers that align with the user's interests by analyzing the query and returning a set of relevant papers. By pulling relevant keyphrases from the keyphrase networks, this was possible. A novel hybrid approach was introduced, which combined co-occurrence and co-citation networks based on their unique connections. This hybrid method improved performance by making the user's query bigger and giving each keyphrase in the query set a certain amount of weight. The combination of co-citation and co-occurrence relations in the proposed method was able to capture co-occurring keyphrases with semantically similar keyphrases to the user's query. The results showed that when the top 40 or 50 articles were chosen for the user's query, the results were more relevant because the proposed method could capture more aspects of the user's query than traditional single network-based methods.

Keywords: Co-occurrence Network, Co-citation Network, Content-based Recommended System, Keyphrase Extraction

Correspondence:
Mayur Makwana
Department of Computer Science and Engineering, Sardar Vallabhbhai National Institute of Technology, Surat-395007, Gujarat, INDIA.
Email: ds18co003@coed.svnit.ac.in
ORCID: 0009-0002-1763-4918

Received: 22-05-2023;
Revised: 08-12-2023;
Accepted: 21-03-2024.

INTRODUCTION

Recommendation systems predict user preferences based on their inputs. It has become a prevalent tool in various domains, such as e-commerce, social media, and mobile commerce, to provide personalized experiences to users. Many novel recommendation algorithms have been developed with technology advancements, making the tool more sophisticated and effective. This paper explores the potential of recommendation systems for academic literature. There are three main types of recommendation techniques: Content-Based Filtering (CBF),[1,2] Collaborative Filtering (CF),[3] and Hybrid approaches.[4,5] They have different reasoning, but they all aim to recommend relevant publications to scholars. CF mainly focuses on other users' behaviors or ratings that have similar profiles to the user.[3,5] CBF uses the user's past preferences and personal library to build a model of the user's interests. Then, CBF compares the candidate papers and the user profiles for similarity. It recommends papers with high similarity to the users. Word embedding methods like Doc2vec[2,6] and Bert[7] are used by researchers to compare semantic similarity instead of exact match in CBF. Word embedding methods have good results, but training models for large data sets and storing vectors is still hard. CBF's argument is simple as a classic way of recommendation.

Our goal is to make a content-based recommendation engine for academic publications. Keywords are used as content descriptors in papers, like in a traditional system. Yao *et al.*[8] successfully recommended a product to users using search keywords. But little effort is devoted to using keywords-based queries to improve scholars' recommendations. Many researchers[9-11] have used the word-word co-occurrence graph G (V, E) to find the relations between keywords. In a graph, V is the set of vertices that are keywords from the literature, and E is the set of edges that connect the vertices if keywords appear together in the same document (E={<vi, vj> |vi, vj ∈ V}). It is also commonly used in keyword-based recommendation systems. Co-word networks often have many general keywords because they consider the number of co-occurrence rather than the importance of each word. The interaction between different publications, such as citation relationships, is not considered; it only focuses on keywords in the same article.[18] Previous research has suggested that citation relationships imply topical relationships between papers.[14,15]

DOI: 10.5530/jscires.13.1.18

Since keywords can directly communicate the topics and core ideas of a paper,[10,12,16] it is reasonable to think that the keywords of citing and cited papers share high semantic similarity.[30] To analyze a semantic relationship between words, we augment the conventional keyword co-occurrence relationship by fusing it with the citation relationship. In recommendation systems, there may be instances where users' keyword entries are incomplete or unclear. The recommended papers should cover the user's specific keywords[19] and have a direct or indirect correlation[21] between a candidate paper with exact query keywords and other papers with numerous relevant keywords. A keyword network has been developed to find keywords relevant to the user's query. The final step is to rank the documents produced by the extended query.

Figure 1 depicts a proposed architecture for a key-phrase-based recommendation system. Key phrases are prioritized over single word keywords when searching scientific publications. Currently, only 2-gram words are used for networks and all other experiments. Based on the user's query, an extended query set is constructed using key-phrase networks to provide satisfactory recommendations. Candidate papers are gathered from the keyphrase-paper association matrix based on an expanded query set. The algorithm then generates ranked target papers. The system is built on three fundamental steps: Keyphrase network creation, expanded query-set, and ranking of scholarly literature. The methodology part discusses different kinds of key-phrase networks, how to choose key-phrases from the network to build a query set, and how to choose weights for ranking. The result section compares the key-phrase co-occurrence network and key-phrase co-citation network and includes corpus building and discussion about recommendations using the co-occurrence network, co-citation network, and combined network.

Related Work

Content-based filtering identifies associations between items based on their content. Important keyword descriptions are frequently utilized to provide an abstract understanding of the entire body of content. When a user searches for a document using a sentence or keyword query, the query's keywords are used to recommend other documents that include the relevant query content. Yao *et al.*[8] suggested a method for recommending products to users by employing keywords, which outperformed existing techniques. Felice *et al.*[22] presented a Keyword-Based Document Recommender System, while Kwang *et al.*[23] created the user profile and generated customised recommendations using keywords.

Co-word Network

Co-word Network Callon *et al.*[13] initially introduced co-word analysis to broaden the scope of co-citation analysis. It is claimed that co-word analysis can permeate the literature and be used to understand the connection between the word and the evolution of a discipline. According to the co-word analysis principle, if two academic terms that are capable of expressing the topic of a specific research area appear in a single article at the same time, there should be some internal relationship between them; specifically, the more frequently they appear in pairs, the closer their relationship.

Research trends and conceptual frameworks can be assessed by calculating the correlation strength between words.[24] Co-word analysis has been effectively used in various fields like informatics,[10,11] recommendation systems,[16] IoT, digital libraries, etc. However, it often focuses on quantity over quality of keywords, leading to a network filled with generic keywords. Standard co-word analysis only considers keywords in the same article, ignoring connections between different articles.[25] This means topically related keywords aren't connected if they aren't in the same article. To improve this, researchers have enhanced the co-word network by using different types of word weights, making the approach more efficient.

Li *et al.*[26] defined a weighted co-occurring keywords time gram and utilized it as a basic unit to analyze temporal information in an existing keywords collection. Second, it can be improved by considering semantic relations. For example, Wang *et al.*[25] proposed a semantic-based co-word analysis that can successfully integrate experts' knowledge into the co-word analysis. Feng *et al.*[27] combined semantic distance measurements with concept matrices generated from ontologically-based concept mapping to improve the co-word analysis method. This study suggests that the co-word analysis method based on semantic distance produces preferable research situation regarding matrix dimensions and clustering results. Despite this method's displayed advantages, it has two limitations: first, it is highly dependent on domain ontology; second, its efficiency and accuracy during concept mapping progress merit.

Keyword Network with Citation Information

Bornmann *et al.*[28] presented a method based on citation context. Words around a citation provide crucial information about

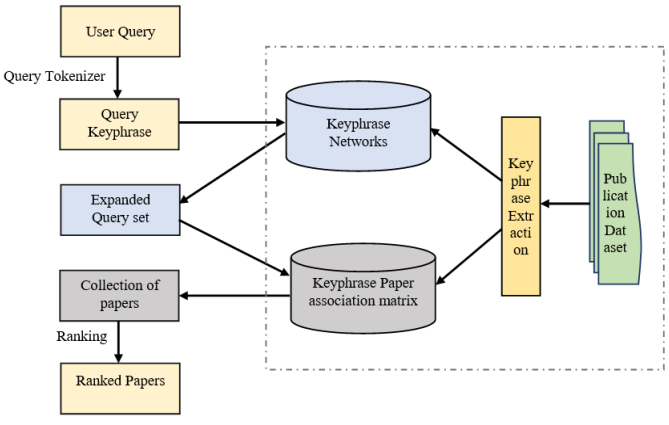

Figure 1: Proposed Key-phrases based Recommendation System Architecture.

its context. During Eugene Garfield's (EG's) lengthy career as information scientist, he published about 1500 papers. The author constructed two networks to analyze and compare the results: co-occurrence networks based on title and abstract keywords from (1) EG's papers and (2) articles mentioning EG's publications. The analysis reveals that papers of EG and citation contexts of cited papers are more semantically related than the titles and abstracts of cited papers. For knowledge discovery, Ding *et al*.[17] created the entity metrics technique, which uses an entity network and assumes that there is some topical relatedness between two articles if one cites the other. Therefore, a keyword in an article and a keyword in its citing article are more topically connected. Cheng *et al*.[32] extended the citation relationship from the article level to the keyword level and created a network of keyword pairs that were implicitly connected by the citation. The main research issue in this study was community detection. Based on the literature review, keyword occurrences and the extended citation relationship between documents and keywords are equally essential for keyword-based analysis. In earlier works, the author used citation relationships for community detection. Fewer attempts are made to use citation relationships for keywords or to combine them with co-occurrence.

Information retrieval models[20] are a fundamental part of many recommendation systems that aim to select and rank relevant documents with respect to a user's query. The proposed system can be seen as an extension of these models, particularly the Classical IR Model, which includes Vector-space, Boolean and Probabilistic IR models. These models represent document contents by a set of descriptors, known as terms, that belong to a vocabulary. The proposed system goes beyond standard information retrieval models by not only matching the user's query with documents but also expanding the query with relevant keyphrases and ranking the documents based on the weights of the keyphrases. In this paper, the goal is to construct a hybrid network by using co-occurrence and co-citation links between nodes to expand the user query to provide more valuable recommendations.

Key-phrases Extraction Techniques

The dataset includes citation, abstract, and title information. High-quality keyword data is crucial for the proposed strategy. Key phrases are prioritized over keywords while searching scientific publications. In this research, keyphrases extracted from the article's title and abstract are used to convey its meaning and aid in developing a robust network. Popular keyword extraction techniques like RAKE, YAKE,[30] TextRank, Summarization, and KeyBert, were reviewed. RAKE and YAKE are two main statistical methods that rely mainly on the frequency and co-occurrence of individual words. Both are quite helpful at recognizing multi-word expressions. YAKE employs additional features, including Casing, which gives capitalized words more weight, and Word Relatedness to Context, which measures how closely a word is related to its context. However, both methods also produce a large number of pointless keyphrases. Therefore, embedding-based techniques Summarization and KeyBert were also reviewed. The T5 model is a Sequence-to-Sequence model that is completely capable of completing any text-to-text tasks. A pre-trained T5-base model generates a summary of the abstract. After the elimination of pause words, the rest of the phrases can be used as keywords by the system. KeyBert finds the terms within a document most similar to the paper by employing BERT-embeddings and basic cosine similarity. KeyBERT is not a one-of-a-kind tool; it was designed to be a quick and easy way to generate keywords and key phrases. It is possible to generate high-quality keyphrases using a variety of permutations on these methods. In order to construct the dataset, the proposed method combined the outputs of keyBert, the T5 model, and KeyBert.

Keyphrase-based Literature Recommendation: Enhancing User Queries with Hybrid Co-citation and Co-occurrence Networks

The proposed strategy combines co-citation and co-occurrence data to form a hybrid network, incorporating elements from both the keyphrases co-citation network (KCN) and the keyphrases co-occurrence network (KCON). This network is constructed for each user input, with KCN nodes central and KCON nodes peripheral.

Figures 2 and 3 show the algorithm and process diagram for the proposed method. It focuses on building networks and processing user queries. Two networks, the Keyphrase Co-occurrence Network (KCON) and the Keyphrase Co-citation Network (KCN), are created from keyphrase data. Once these networks are built, the system can process user queries, which can be key phrases or sentences. For example, if a user submits a query with two key phrases, q1 and q2, the system creates a hybrid network for q1. It selects top keyphrases from the KCN as core nodes, and for each core node, it picks top keyphrases from the KCON as outer nodes. This process is repeated for q2. After forming this hybrid structure, the system assigns weights to all nodes. This section explains this procedure in detail.

Key-phrases Networks

Key-phrases Co-occurrence Network (KCON)

As a type of content analysis, co-occurrence analysis measures the degree of association between key terms in a corpus. The relationship between terms becomes stronger the more often they appear together. Two nodes, V_{ki} and V_{kj}, are connected by the edge E_{kij} ($E_{kij} \in E_k$) if at least one article has both keyphrases corresponding to these two vertices. The number of articles containing both the keyphrases corresponding to V_{ki} and V_{kj} determines the weight of W_{kij} linked with an edge E_{kij}. A sample of how a network is constructed is shown in Figure 4(A). The relationship between K1 and K2 has a weight of 1 since they appear together only once in paper P1. Since K3 and K4 are mentioned

	Algorithm 1: Proposed Scholarly Literature Recommendation System
	***Input**: Query Key-phrases, Keyword co-citation network (KCN), Keyword co-occurrence network (KCON)*
	***Output**: Ranked Documents*
1	***Initialization of variables*** *Q <- Query Key-phrases* *S <- Query Sets*
2	***For** every element in Q*
3	*K <- Select Top Key-phrases from KCN for Q_i*
4	***For** every element in K*
5	*C <-Select Top Key-phrases from KCON for K_j*
6	***End For***
7	*S_i <- Add Q_i, K and C to set*
8	*Assign weights for every key-phrases in S_i*
9	***End For***
10	*Rank documents based on common key-phrases and total weight*

Figure 2: Algorithm of Proposed Scholarly Literature Recommendation System.

twice in the aforementioned documents, the link between them carries twice as much significance as any other.

Key-phrase Co-citation Network (KCN)

The traditional co-word analysis only focuses on keywords that appear in the same article, without considering the relationship between different articles;[32] therefore, we are working on a key-phrase citation network to take topic relatedness into consideration. A citation network is defined as a graph G =<V, E> where each node $V_i \in V$ represents a Keyphrase and edge E_{ij} from V_i to V_j indicates that the keyphrase representing V_j cites the keyphrase representing V_i or vice-versa. Figure 4(B) shows the example of the network construction. Paper P1 cites P2, so every keyphrase of paper P1 links to every keyphrase of P2. The link between K1 and K2 is having 3 weights because three common neighboring nodes exist between K1 and K2. The keyword citation graph has a greater number of connections, and it is a well-connected graph compare to co-word graph.

Query Expansion

The term "Expanded query-set" refers to a set of search terms related to the original query. KCN is known for finding similar keywords. Our analysis (section 4.3) shows that KCN is closer to semantically relevant key phrases than KCON, which returns

frequently occurring words. Tracking this data is also important as it shows popular approaches to the topic. We first collected most relevant key phrases from KCN for the user's query, then did the same for KCON. This process gathered semantically related and co-occurring phrases. Section 4.5 analyzes the results of using KCN, KCON, and both together. Here, user can select the top three or four key terms from each network. Tables 1 and 2 show an example of the query expanding process.

Weight Selection for Expanded Query-set

The Expanded Query-Set provides insights into a user's query, but not all phrases in the set are equally important. To ensure the recommendation system displays the most relevant documents first, we assign weights to keyphrases based on their significance. We collected keyphrases in two phases and assigned a total weight of 1 to each phase, resulting in a total weight of 2 for each query set. Before assigning weights, we followed several rules:

1. Primary key phrases (user query) should have greater weight.
2. KCON and KCN rankings are based on co-occurrence and co-citation values.
3. Higher-ranking KCN co-occurrence phrases should be given more weight than lower ranking KCN keyphrases.
4. In the case of redundancy, choose the highest weight for the key phrase.

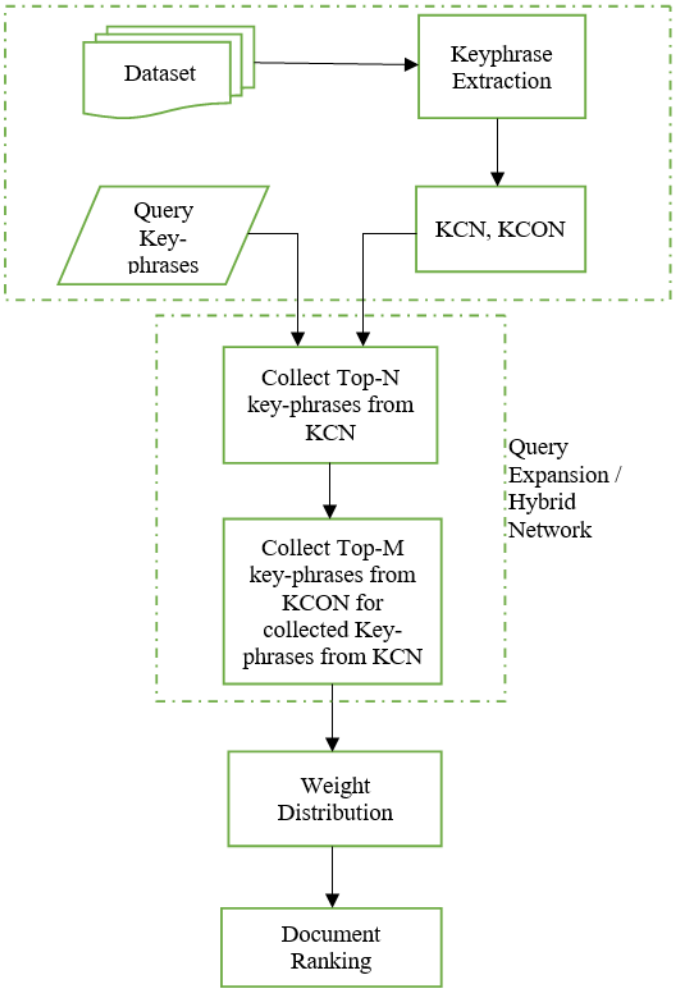

Figure 3: Process Diagram of Proposed Scholarly Literature Recommendation System.

We used a weight propagation method for weight distribution, where KCON keyphrase gets weight from KCN keyphrase. An example of weight distribution is shown in Table 3 using the algorithm in Figure 5. For the query "Concurrency control", the top three key phrases are "Distributed database," "Database System," and "Control Algorithms". The total weight of 1 is divided into four parts: 0.4, 0.3, 0.2, and 0.1. The Phase-II results for 'Concurrency control' from KCON yield the top three key phrases with temporary weights of 0.5, 0.3, and 0.2. However, according to the rules of weight propagation, the combined value of these weights should be 0.4. Thus, the final weight is obtained by multiplying temporary weights by the "Concurrency control" weight of 0.4. The computation is displayed in Table 3, and the resulting query set with ranked keyphrases is displayed in Table 4.

Implementation and Results

This section is about experiments performed on a scholarly literature dataset like data sampling, key-phrase extraction, network building and recommendation.

Dataset

Data are essential for the experiments of the recommendation methods. We chose the dataset published by Hung N. *et al.*[29] for our study due to its comprehensive metadata. Although there are several publicly available datasets for scientific publication recommendations, such as DBLP, PubMed, CiteSeerX, and Microsoft Academic Graph, each has its own limitations. DBLP does not provide abstracts and keywords, PubMed lacks keywords, CiteSeerX does not include citation information, and Microsoft Academic Graph is not freely accessible. In contrast, the dataset by Hung N. *et al.*[29] includes all necessary metadata fields: title, abstract, keywords, and citation information.

After removing the papers with missing titles, abstracts, and citation information, Dataset has 702,643 papers published from 1965 to 2009 with 7,654,677 citation information. Each paper is then managed by (a) fetching its abstract and title; (b) removing

Table 1: Phase-1, Key-phrases from KCN.

User Query = Concurrency Control	Top - 3 Key-phrases from KCN
	Distributed Database
	Database System
	Control Algorithms

Paper	Key-phrases			Citation
P1	K1	K2		P1->P2
P2	K2	K3	K4	P2->P3
P3	K4	K3	K5	P3->P1

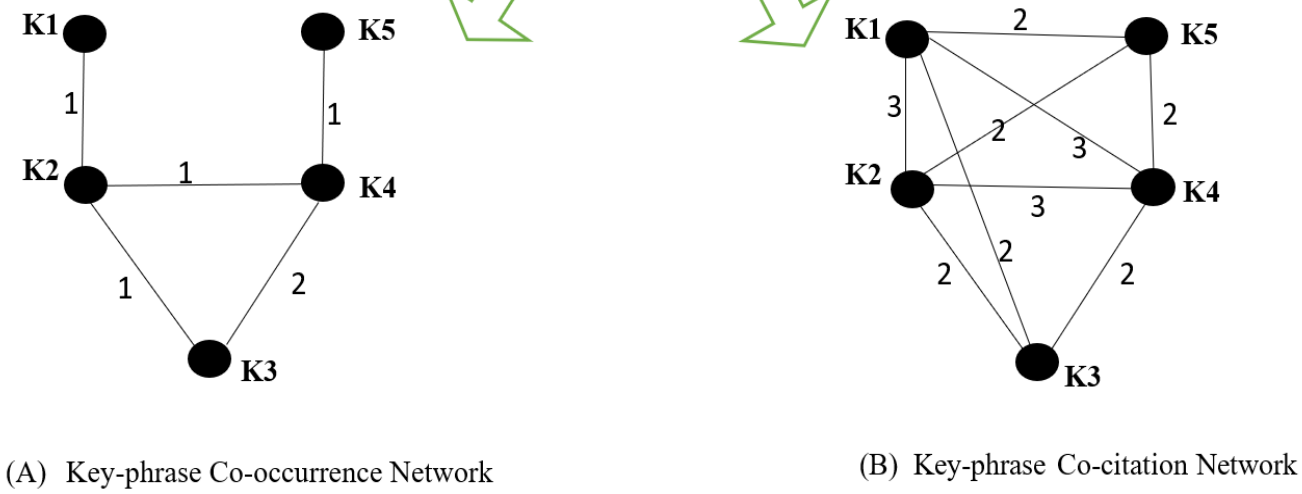

Figure 4: Key-phrases Networks (A, B).[42]

Table 2: Phase-2, Key-phrases from KCON.

Phase-I Key-phrase	Concurrency Control	Distributed Database	Database System	Control Algorithms
Top – 3 key-phrases from KCON	Control Algorithms	Database System	Relational Database	Concurrency Control
	Database Concurrency	Concurrency Control	Database Management	Database Concurrency
	Distributed Database	Transaction Processing	Optimization algorithms	Shared Database

the noisy words, and (c) fetching citation information. Due to its huge size, it is difficult to perform experiments with limited resources.

We needed to select a data sample with good citation connectivity. By selecting random documents, we might lose the citation information. The proposed approach is highly dependent on the citation network. So, we followed a simple process. We chose one random document and created a network up to the depth of 4 by considering each citation link has the same value as 1. We collected 7,433 documents from this process. After removing redundancy and papers with less than five citations, we got 5,761 papers.

We extracted keyphrases from 5761 papers, yielding over a hundred thousand keywords. Details regarding our dataset are presented in Table 5. The retrieved keyphrases have several unrelated Key-values with strange keyword combinations and various variants of Key-values with the same root. Redundancy can be avoided by storing the essential phrases following stemming. In addition, the problem with the uncommon keyword combinations was resolved by following a few easy procedures. There is a good chance that the uncommon keyphrases will not appear in many other documents because they are specific to the documents themselves. While analysing the Keyphrase Co-occurrence network, it came to our attention that most of these uncommon key phrases behave as outliers in a network.

Table 3: Weight Distribution.

Concurrency Control		Distributed Database		Database System		Control Algorithms	
0.4		0.3		0.2		0.1	
Control Algorithms (0.5)	(0.5*0.4) 0.20	Database System (0.5)	(0.5*0.3) 0.15	Relational Database (0.5)	(0.5*0.2) 0.10	Concurrency Control (0.5)	(0.5*0.1) 0.05
Database Concurrency (0.3)	(0.3*0.4) 0.12	Concurrency Control (0.3)	(0.3*0.3) 0.09	Database Management (0.3)	(0.3*0.2) 0.06	Database Concurrency (0.3)	(0.3*0.1) 0.03
Distributed Database (0.2)	(0.4*0.2) 0.08	Transaction Processing (0.2)	(0.3*0.2) 0.06	Optimization algorithms (0.2)	(0.2*0.2) 0.04	Shared Database (0.2)	(0.2*0.1) 0.02

After steaming and removing outliers, the corpus size was reduced to approximately 62 thousand.

Analysis of Key-phrase Extraction Techniques

We decided to employ precision, coverage of the keyphrases given by the authors in our analysis. We employed the Python NLTK suite for pre-processing, and we used the KeyBert and T5 models for embedding. Moreover, in statistical approaches, YAKE and RAKE have GitHub implementations. Table 6 shows Coverage, and precision for the top-10 extracted keyphrases.

Table 6 demonstrates that KeyBert and YAKE outperform the other approaches. In addition, RAKE has the lowest coverage and precision values among all approaches. TextRank is comparable to RAKE in performance. KeyBert and the T-5 model, in particular, outperform alternative approaches in terms of precision. In our analysis, statistical techniques had a greater coverage but poor precision. Coverage values can be improved by expanding the number of candidates to 20 instead of 10. e.g., RAKE performs better in the Top 20. However, the Top-10/20 value for the threshold may have an impact on customer satisfaction. The low threshold could lead to more irrelevant keywords and keyphrases, decreasing precision while increasing coverage. However, a significant threshold may lower the coverage while maintaining acceptable precision. According to our statistical method analysis, they are simpler and more efficient for large amounts of data; however, they produced more irrelevant key phrases for small amounts of data. When it comes to getting keyphrases out of short texts, graph-based and embedding methods worked better than statistical methods. Nonetheless, while assessing the candidates' keyphrases for several publications, we discovered that KeyBert frequently overlooks unique or off-topic keyphrases. The embedding method, like KeyBert, concentrates more on centre concepts. The top-N keyphrases produced by KeyBert are those that most closely resemble documents. A statistical technique such as YAKE could easily find unique or multi-disciplinary key terms. As a result, we combined the keyphrases from KeyBert, YAKE, and tokens from the document titles to create a corpus of keyphrases. Furthermore, we eliminated the redundant and unnecessary keyphrases produced by YAKE using outlier removal from the co-occurrence network.

Key-phrases Co-citation Network (KCN) vs Key-phrases Co-occurrence Network (KCON)

KCN and KCON Analysis

In query-set building, we picked the top keyphrases first from the KCN and then from KCON. The prevalent question is why the recommended strategy followed this sequence. To address this query, it is necessary to understand the distinction between KCN and KCON. In building the query-set, keyphrases were selected from KCN first, then from KCON. To explain this choice, we need to understand the difference between the two networks. Seven characteristics were examined to determine the difference between co-occurrence and co-citation networks: nodes, edges, Average Degree (AD), Network Diameter (ND), Average Clustering Coefficient (ACC), Average Path Length (APL), and Modularity (M). The word2vec embedding technique was also used to assess semantic similarity between adjacent nodes. ND represents the longest shortest path in a network, serving as a measure of the network's linear size. ACC characterizes the degree of cooperation between a node's neighbors, indicating how well the nodes in the network cluster together. Modularity is a common method for assessing the robustness of a network's community structure.[30,32] It is often used to divide a network community to identify research topics.[34] Calculations for these parameters were performed using Gephi[33] and NetworkX. Table 7 presents the statistical results. Despite having the same number of nodes (63,985), KCN has nine times as many edges (63,985,00). The higher AD of KCN indicates it is denser than KCON. In KCN, citation relationships reduce the distance between nodes, altering typical values for degrees, clustering indices, and path lengths. According to both ACC and modularity, the co-occurrence network has superior cluster detection quality. Gephi identified 63 partitions in KCON and 35 in KCN. Figure 6 displays the clusters found in KCN and KCON.

While manually analyzing the surrounding nodes in the two networks, we discovered that KCN maintains similar nodes closer together than KCON. Additionally, it seen by comparing the word2vec vectors. Tables 8 and 9 detail the obtained outcomes. For the better result, keyphrases were preserved as multi-word phrases during the word2vec model training. Cosine similarity between surrounding nodes is better in KCN than in co-occurrence networks, as the top nodes are more closely connected to one another. Therefore, we first retrieve the keyphrases from KCN, and then retrieve popular terms from KCON for each phrase.

Network Data-structure

The system constructs the Query-set by mining related keyphrases from co-citation and co-occurrence networks in response to a user query. About 70.000 key phrases are available. With such few resources, it is difficult to store this much data in the matrix. A hashmap containing a hashmap of lists has been created and kept in this application.

Figure 7 shows how two-tiered hash maps or hash tables are used. The first level hashmap has all the unique keyphrases in the corpus. The entry "Machine Scheduling" in level-1 hashmap maps to the level-2 hashmap, which has the key terms that co-occur with Machine Scheduling (e.g., Scheduling algorithms). Each element in the second hashmap leads to a list of papers with these two key terms (e.g., machine scheduling and scheduling algorithms). The dictionary name for KCON is Keyp co_occ. By running Kep co_occ [X] [Y], the system gets all common documents with X and Y together. A similar dictionary for the Co-citation network (Keyp_co_c) is built. By running Keyp_co_cita, the number of shared links or affinities can be obtained.

Query-set and Ranking

Traditionally, researchers have used a co-occurrence network. However, as we saw previously, a co-citation network has some advantages over co-occurrence when it comes to identifying semantically similar phrases. For instance, the co-occurrence network may show a strong relationship between the terms "decision tree" and "classification" due to their frequent occurrence.

Table 4: Final Query-set.

Rank	Key Phrase	Weight
1	Concurrency Control	0.4
2	Distributed Database	0.3
3	Control Algorithms	0.2
4	Database System	0.2
5	Database Concurrency	0.12
6	Relational Database	0.1
7	Transaction Processing	0.06
8	Optimization algorithms	0.04
9	Shared Database	0.02

Table 5: Statistics of Dataset.

No. of Documents	5761
No. of Citation Links	53340
No. of Extracted Key-phrases	1,02,386
No. of Key-phrases after removing redundancy	72,560
No of Key-phrases after removing outliers	62423

Table 6: Evaluation of Key-phrases Extraction algorithms.

Algorithm	Precision	Coverage
RAKE	0.45	0.5
YAKE	0.5	0.64
Summarization	0.64	0.4
TextRank	0.45	0.55
KeyBert	0.6	0.6

Algorithm 2: Weight selection for Query-Set

Input: *Query Key-phrase*

Output: *Query-Set with weights*

1 ***Initialization of variables***
Q <- Query Key-phrases
S <- Query Sets
W<- Key-phrases Weights

2 ***For*** *every Key-phrases in Q*

3 *K <- Q_i + Top-3 Key-phrases from KCN for Q_i*

Assign W[k1] = 0.4, w[k2] =0.3, w[k3] =0.2, w[k4] =0.1

4 ***For*** *every Key-phrases in K*

5 *C <-Select Top-4 Key-phrases from KCON for K_i*

6 *Assign Temporary weights 0.4, 0.3, 0.2, 0.1 to T[1 to 4] for K[1 to 4]*

7 ***For*** *every Key-phrases in C*

8 ***If*** *($W[C_i] < W[K_i] * T[i]$)*

9 *Assign final weight $W[C_i] = W[K_i] * T[i]$*

10 ***End IF***

11 ***End For***

12 ***End For***

13 ***End For***

Figure 5: Algorithm for Proposed Weight Selection.

Regardless, these terms are not synonymous. Nevertheless, it is also essential because it is one of the most popular classification methods. The results of our trials led us to settle on the idea of using the first Co-citation network to compile query terms that share a common semantic ground. After collecting a set of keywords, we used a co-occurrence network to identify the most important associated phrases. Therefore, the query returns related and popular key terms associated with the user's query.

After constructing a query set, the next critical step is assigning relative weight to each keyphrase. The process is described in the prior section. Table 10 displays the extended query set and weights for the user's query. The top 10 documents for the query set are displayed in Table 11, along with their scores and the number of common key phrases. Here the total weight is the sum of the common keyphrases with an extended query set, and the maximum possible weight is 2. Tables 12 and 13 show

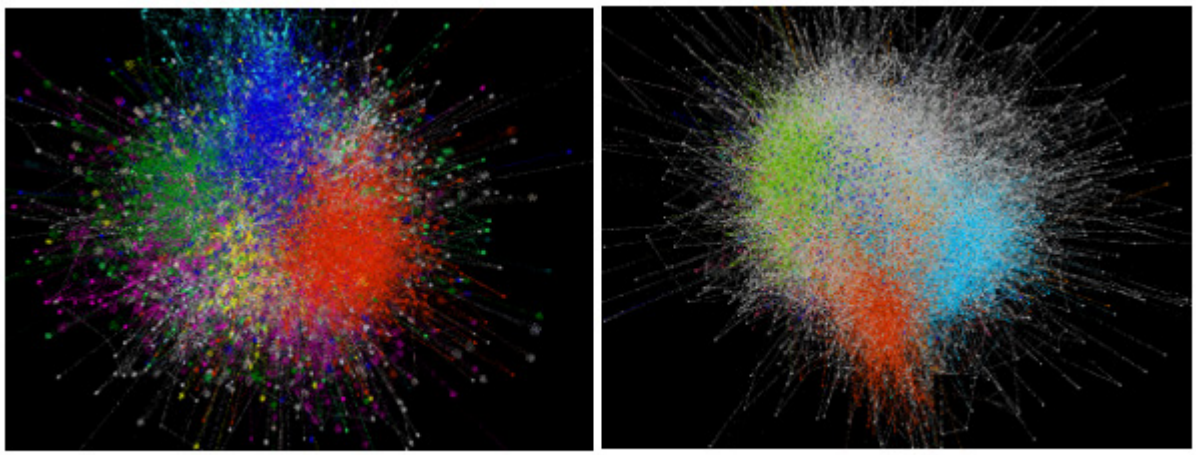

Figure 6: Visualization of KCN (A) and KCON (B) using Gephi.

Table 7: Network Analysis.

	No. of nodes	No. of links	Avg. Degree	Avg. path length	ND	ACC	Modularity
KCON	63985	764426	23	7.65	8	0.9	0.726
KCN	63985	6398500	201	4.2	6	0.3	0.486

Table 8: Vector Comparison between Top Neighbouring Nodes in KCON.

	Mining Algorithms (1)	Machine Learning (2)	Large Databases (3)	Knowledge Discovery (4)	Time Series (5)
Data Mining	0.75	0.4	0.64	0.48	0.42

Table 9: Vector Comparison between Top Neighbouring Nodes in KCN.

	Association Mining (1)	Mining Algorithms (2)	Large Databases (3)	Association Rules (4)	Knowledge Discovery (5)
Data Mining	0.69	0.75	0.64	0.54	0.48

Table 10: Query-sets using Different Approaches.

User Query-Association rule			
Co-occurrence	Co-citation	Co-citation (Top-3) + Co-occurrence (Top-4)	
TOP-5	TOP-5	Key-phrases	Weight
Association rule	Association rule	Association rule	0.4
Association mining	Association mining	Association mining	0.3
Large databases	Data mining	Data mining	0.2
Mining algorithms	Large Database	Mining algorithms	0.12
Efficient Algorithm	Knowledge Discovery	Large Database	0.1
Data mining	Frequent pattern	Efficient Algorithm	0.08
		Machine Learning	0.08
		Knowledge discovery	0.06
		Frequent pattern	0.03
		Database System	0.02
		Pattern mining	0.01

Table 11: Top-10 Documents using Co-occurrence and Co-citation Query-set.

	Query -> Association rule (Co-occurrence + Co-citation)		
No	**Paper Title**	**Total weight**	**Common keyphrases**
1	Evaluation of Sampling for Data Mining of Association Rules	0.71	3
2	An Effective Hash Based Algorithm for Mining Association Rules	0.71	3
3	Scalable Algorithms for Association Mining	0.64	4
4	An Efficient Algorithm for Mining Association Rules in Large Databases	0.62	3
5	Mining Association Rules with Item Constraints	0.58	3
6	Mining association rules between sets of items in large databases	0.52	3
7	Mining Generalized Association Rules	0.5	2
8	Discovery of Multiple-Level Association Rules from Large Databases	0.5	2
9	Sampling Large Databases for Association Rules	0.48	2
10	ITL-MINE: Mining Frequent Itemsets More Efficiently	0.48	2

Table 12: Top-10 Documents using Co-occurrence.

	Query -> Association rule (Co-occurrence)	
No	**Paper Title**	**Common keyphrases**
1	Evaluation of Sampling for Data Mining of Association Rules.	3
2	An Efficient Algorithm for Mining Association Rules in Large Databases.	3
3	An Effective Hash Based Algorithm for Mining Association. Rules	3
4	An efficient association mining implementation on clusters of SMP	3
5	Scalable Algorithms for Association Mining.	3
6	Static Versus Dynamic Sampling for Data Mining.	3
7	Mining association rules between sets of items in large databases.	3
8	Mining Generalized Association Rules.	2
9	ITL-MINE: Mining Frequent Item sets More Efficiently.	2
10	Sampling Large Databases for Association Rules.	2

the top 10 documents that were found using the independent Co-occurrence network and the Co-citation network query sets.

DISCUSSION

In this study, the performance of a document recommendation system was evaluated using various query sets. The process of selecting the top-N keyphrases from KCN or KCON to construct the query sets was identified as a crucial factor in the system's effectiveness. If a small value of N is chosen, some important keyphrases may be missed, but selecting a large value may result in redundant keyphrases that negatively impact the output.

The proposed approach leverages the synergistic effects of KCN and KCON, allowing for the possibility of capturing the majority of important keyphrases, even with a lower value of N. This was validated through the results obtained in Tables 4 and 10, which showed that good quality query sets were generated with even lower values of N. The results also indicated that the proposed method, using both KCN and KCON, was more effective in capturing the user's query than using either network alone.

A good recommender system exhibits keywords overlap between query set keywords and recommended paper. The obtained results include seven papers sharing three key phrases with the KCON extended query set, refer in Table 12. On the other hand, there are four documents in Table 13 which are having three common key phrases with the KCN extended query set. Authors are less likely to use semantically similar keyphrases together in a paper. This result is one evident output of it and therefore there are not many use cases of independent keyphrase citation network in recommendation. But semantically similar keyphrases can help in gathering slightly similar papers to the user's actual query which was visible in output. For example, when the top 40 or 50 papers were selected for user's query, obtained results were more

Table 13: Top-10 Documents using Co-citation Query-set.

	Query -> Association rules (Co-citation)	
No	**Paper Title**	**Common keyphrases**
1	Evaluation of Sampling for Data Mining of Association Rules.	3
2	An Effective Hash Based Algorithm for Mining Association Rules.	3
3	Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach.	3
4	Scalable Algorithms for Association Mining.	3
5	H-Mine: Hyper Structure Mining of Frequent Patterns in Large Databases.	2
6	Mining Generalized Association Rules.	2
7	Discovery of Multiple-Level Association Rules from Large Databases.	2
8	An Efficient Algorithm for Mining Association Rules in Large Databases.	2
9	Sampling Large Databases for Association Rules.	2
10	An efficient association mining implementation on clusters of SMP.	2

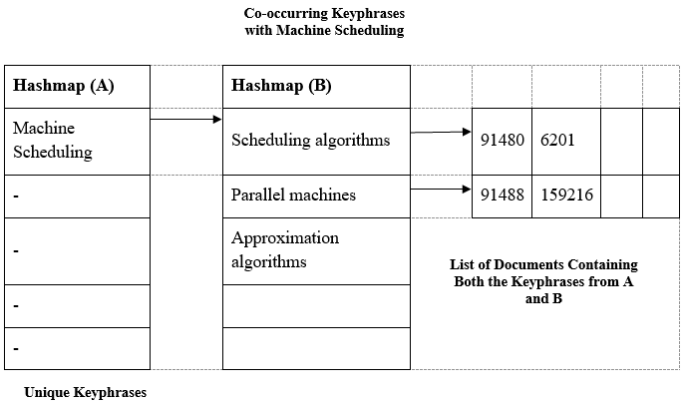

Figure 7: Data Storage.

relevant than using either KCN or KCON. Our approach has demonstrated the ability to capture multiple facets of the user's query, and this is a unique advantage of our method over existing solutions.

Library Services Platforms (LSPs) represent a specific type of resource management system. They are designed to manage and provide access to a diverse range of library resources, including books, journals, and databases. Our proposed system could significantly enhance the search functionality of LSPs. Unlike existing LSPs, which primarily rely on simple keyword matching for search functionality, our proposed system employs a hybrid approach. This approach integrates co-occurrence and co-citation networks to enrich the user's query, thereby retrieving more diverse and relevant results than a simple keyword-based search. If the LSP allows for user profile building, the proposed system can be particularly beneficial. As part of active learning or to address the cold start problem, the system can initially ask users about their interests. Based on this information, the proposed approach can provide suggestions related to the user's interests. This process may occur multiple times, enhancing its effectiveness over time. Not only will this help generate personalized recommendations, but the system's keyphrase or topic suggestions could also help expand the user's vision. This could be instrumental in extending their knowledge base. Consider an LSP used by a university library where a researcher is seeking papers on 'machine learning in healthcare'. Our proposed system presents the researcher with a list of recommended papers that not only cover 'machine learning' or 'healthcare', but also various aspects of 'machine learning in healthcare'. This could include papers discussing different machine learning techniques used in healthcare, applications of machine learning in various healthcare areas, and the challenges and opportunities associated with using machine learning in healthcare.

The proposed approach also has some limitations, such as the generation of redundant keywords in the expanded query sets. This limitation needs to be addressed in future research, and it can be achieved by selecting distinct keywords while generating the query set. The challenge of selecting the Top-N keyphrases from KCN or KCON, especially when many keyphrases have the same Co-occurrence or Co-citation count, is another area for future exploration. In addition, it would be interesting to investigate other keyphrase extraction algorithms, instead of KeyBert and YAKE, to further improve the performance of the recommender system. And, YAKE is known to produce many irrelevant keyphrases, so finding ways to effectively filter these out while retaining distinctive, crucial keyphrases will be an important challenge for future research in this field.

CONCLUSION

In summary, our study aimed to enhance the understanding of the relationship between articles and keywords by introducing a novel approach to keyword citation analysis. By combining the strengths of co-occurrence and co-citation networks, we have produced meaningful results, but there is still work to be done to improve the accuracy and efficiency of the method. In future research, we plan to validate our findings using different datasets, address the challenges of key-phrase extraction, and find a more scalable solution for processing large volumes of data in a distributed environment. The results of this research will contribute to a better understanding of the relationships between keywords in academic literature and have implications for a range of applications, including bibliometrics, knowledge management, and text analysis.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

REFERENCES

1. Shahbaz A, Muhammad TA. Combining metadata and co-citations for recommending related papers. Turk J Elec Eng Comp Sci. 2020; 28(3): 1519-34 TÜBİTAK doi:10.3906/elk-1908-19.
2. Ganguly S, Pudi V. Combining graph and text information for scientific paper. Representation. 2017: 383-95.
3. Haruna K, Akmar Ismail M, Damiasih D, Sutopo J, Herawan T. A collaborative approach for research paper recommender system. PLOS ONE. 2017; 12(10): e0184516. doi: 10.1371/journal.pone.0184516, PMID 28981512.
4. Rodriguez-Prieto O, Araujo L, Martinez-Romo J. Discovering related scientific literature beyond semantic similarity: a new co-citation approach. Scientometrics. 2019; 120(1): 105-27. doi: 10.1007/s11192-019-03125-9.
5. Hassan HAM. Personalized research paper recommendation using deep learning; 2017. doi: 10.1145/3079628.3079708.
6. Le Q, Mikolov T. Distributed representations of sentences and documents. In:International conference on machine learning, 2014. PMLR, 1188-1196..
7. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L. Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805. 2018.
8. Yao J, Yao J, Yang R, *et al.* Product recommendation based on search keywords. In: Ninth Web Information Systems and Applications Conference. IEEE Publications; 2012. p. 67-70.
9. Callon M, Courtial JP, Turner WA, Bauin S. From translations to problematic networks: an introduction to co-word analysis. Soc Sci Inf. 1983; 22(2): 191-235. doi: 10.1177/053901883022002003.
10. Khasseh AA, Soheili F, Moghaddam HS, Chelak AM. Intellectual structure of knowledge in iMetrics: A co-word analysis. Inf Process Manag. 2017; 53(3): 705-20. doi: 10.1016/j.ipm.2017.02.001.
11. Sedighi M. Application of word co-occurrence analysis method in mapping of the scientific Felds (case study: the feld of informetrics). Library Review. 2016;65(1/2):52-64. doi: 10.1108/LR-07-2015-0075.
12. Zhang W, Zhang Q, Yu B, Zhao L. Knowledge map of creativity research based on keywords network and co-word analysis, 1992-2011. Qual Quant. 2015; 49(3): 1023-38. doi: 10.1007/s11135-014-0032-9.
13. Callon M, Courtial JP, Laville F. Co-word analysis as a tool for describing the network of interactions between basic and technological research: the case of polymer chemistry. Scientometrics. 1991; 22(1): 155-205. doi: 10.1007/BF02019280.
14. Bornmann L, Haunschild R, Hug SE. Visualizing the context of citations referencing papers published by Eugene Garfeld: A new type of keyword co-occurrence analysis. Scientometrics. 2018; 114(2): 427-37. doi: 10.1007/s11192-017-2591-8, PMID 29449748.
15. Zhu Y, Song M, Yan E. Identifying liver cancer and its relations with diseases, drugs, and genes: A literature-based approach. PLOS ONE. 2016; 11(5): e0156091. doi: 10.1371/journal.pone.0156091, PMID 27195695.
16. Hu J, Zhang Y. Research patterns and trends of recommendation system in china using coword analysis. Inf Process Manag. 2015; 51(4): 329-39. doi: 10.1016/j.ipm.2015.02.002.
17. Song M, Han NG, Kim YH, Ding Y, Chambers T. Discovering implicit entity relation with the gene-citation-gene network. PLOS ONE. 2013; 8(12): e84639. doi: 10.1371/journal.pone.0084639, PMID 24358368.
18. Wang ZY, Li G, Li CY, Li A. Research on the semantic-based co-word analysis. Scientometrics. 2012; 90(3): 855-75. doi: 10.1007/s11192-011-0563-y.
19. Zhou X, Wu B, Jin Q. Analysis of user network and correlation for community discovery based on topic-aware similarity and behavioral influence. IEEE Trans Hum Mach Syst. 2018; 48(6): 559-71. doi: 10.1109/THMS.2017.2725341.
20. Amati G. Information retrieval models. In: Liu L, Özsu MT, editors. Encyclopedia of database systems. New York: Springer; 2018. doi: 10.1007/978-1-4614-8265-9_916.
21. Liu H, Kou H, Yan C, Qi L. Link prediction in paper citation network to construct paper correlation graph. EURASIP J Wirel Commun Netw. 2019; 2019(1): 233. doi: 10.1186/s13638-019-1561-7.
22. Ferrara F, Pudota N, Tasso C. A keyphrase-based paper recommender system. In: IRCDL. Vol. 249. Center for Comparative Immigration Studies; 2011. p. 14-25. doi: 10.1007/978-3-642-27302-5_2.
23. Hong K, Jeon Hocheol, Jeon C. UserProfile-based personalized research paper recommendation system 8th International Conference on Computing and Networking Technology (ICCNT 2012). Vol. 8(1); 2012. p. 134-8.
24. Yan BN, Lee TS, Lee TP. Mapping the intellectual structure of the Internet of Things (IoT) feld (2000-2014): A co-word analysis. Scientometrics. 2015; 105(2): 1285-300. doi: 10.1007/s11192-015-1740-1.
25. Li S, Sun Y. The application of weighted co-occurring keywords time gram in academic research temporal sequence discovery. Proc of Assoc for Info. 2013; 50(1): 1-10. doi: 10.1002/meet.14505001037.
26. Feng J, Zhang YQ, Zhang H. Improving the co-word analysis method based on semantic distance. Scientometrics. 2017; 111(3): 1521-31. doi: 10.1007/s11192-017-2286-1.
27. Bornmann L, Haunschild R, Sven E. Hug3 Visualizing the context of citations referencing papers published by Eugene Garfield: a new type of keyword co-occurrence analysis. Scientometrics. 2018; 114: 427-37. doi: 10.1007/s11192-017-2591-8.
28. Tran HN, Huynh T, Hoang K. A potential approach to overcome data limitation in scientific publication recommendation. In2015 Seventh International Conference on Knowledge and Systems Engineering. 2015. IEEE, 310-313.
29. Campos R, Mangaravite V, Pasquali A, Jorge A, Nunes C, Jatowt A. YAKE! Keyword extraction from single documents using multiple local features. Information Sciences. 2020; 509: 257-89. doi: 10.1016/j.ins.2019.09.013.
30. Newman ME. Community detection and graph partitioning. Europhys Lett. 2013; 103(2): 28003. doi: 10.1209/0295-5075/103/28003.
31. Cheng Q, Wang J, Lu W, Huang Y, Bu Y. Keyword-citation-keyword network: a new perspective of discipline knowledge structure analysis. Scientometrics. 2020; 124(3): 1923-43. doi: 10.1007/s11192-020-03576-5.
32. Blondel VD, Guillaume J, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. J Stat Mech Theor Exp. 2008; 2008(10): 10008. doi: 10.1088/1742-5468/2008/10/P10008.
33. Bastian M, Heymann S, Jacomy M. Gephi: an open source software for exploring and manipulating networks. In: Proceedings of the international AAAI conference on web and social media. 2009. 3(1):361-362. doi: 10.13140/2.1.1341.1520.
34. Wang X, Cheng Q, Lu W. Analyzing evolution of research topics with NEViewer: A new method based on dynamic co-word networks. Scientometrics. 2014; 101(2): 1253-71. doi: 10.1007/s11192-014-1347-y.

Cite this article: Makwana M, Mehta RG. Keyphrase-Based Literature Recommendation: Enhancing User Queries with Hybrid Co-citation and Co-occurrence Networks. J Scientometric Res. 2024;13(1):217-29.