DiSCern: A Diversified Citation Recommendation System for Scientific Queries

Tanmoy Chakraborty #1, Natwar Modani *2, Ramasuri Narayanam *3, Seema Nagar *4

# *Department of Computer Science & Engineering*
Indian Institute of Technology, Kharagpur, India – 721302
1 its_tanmoy@cse.iitkgp.ernet.in

* *IBM Research, India*
{2 namodani, 3 ramasurn, 4 senagar3}@in.ibm.com

***Abstract*—Performing literature survey for scholarly activities has become a challenging and time consuming task due to the rapid growth in the number of scientific articles. Thus, automatic recommendation of high quality citations for a given scientific query topic is immensely valuable. The state-of-the-art on the problem of citation recommendation suffers with the following three limitations. First, most of the existing approaches for citation recommendation require input in the form of either the full article or a seed set of citations, or both. Nevertheless, obtaining the recommendation for citations given a set of keywords is extremely useful for many scientific purposes. Second, the existing techniques for citation recommendation aim at suggesting prestigious and well-cited articles. However, we often need recommendation of diversified citations of the given query topic for many scientific purposes; for instance, it helps authors to write survey papers on a topic and it helps scholars to get a broad view of key problems on a topic. Third, one of the problems in the keyword based citation recommendation is that the search results typically would not include the semantically correlated articles if these articles do not use exactly the same keywords. To the best of our knowledge, there is no known citation recommendation system in the literature that addresses the above three limitations simultaneously. In this paper, we propose a novel citation recommendation system called *DiSCern* to precisely address the above research gap.**

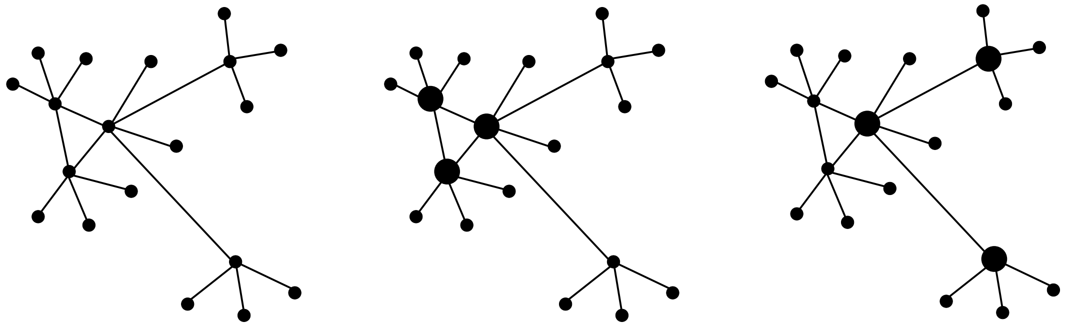
DiSCern finds relevant and diversified citations in response to a search query, in terms of keyword(s) to describe the query topic, while using only the citation graph and the keywords associated with the articles, and no latent information. We use a novel keyword expansion step, inspired by community finding in social network analysis, in DiSCern to ensure that the semantically correlated articles are also included in the results. Our proposed approach primarily builds on the Vertex Reinforced Random Walk (VRRW) to balance prestige and diversity in the recommended citations. We demonstrate the efficacy of DiSCern empirically on two datasets: a large publication dataset of more than 1.7 million articles in computer science domain and a dataset of more than $29,000$ articles in theoretical high-energy physics domain. The experimental results show that our proposed approach is quite efficient and it outperforms the state-of-the-art algorithms in terms of both *relevance* and *diversity*.

I. Introduction

Finding relevant scholarly articles from the literature for a given topic is an important task for several scientific activities. This may be required, for example, to understand the current state-of-art in the topic, or to provide the citations while writing a research article. With more than one hundred thousand new papers published each year, performing a complete literature survey to find relevant articles has become a difficult task for research community. Researchers typically rely on manual methods such as keyword-based search via web search engines, reading proceedings of conferences, and browsing publication lists of known experts in the respective fields. These techniques are laborious as well as time-consuming, and they allow to reach only a limited set of articles in a reasonable time. For these reasons, there has been a significant effort from research community to develop automatic recommendation systems that help researchers to find relevant articles [1]–[3].

While there is a significant body of work on the design of citation recommendation systems, the state-of-the-art on this problem suffers with the following three limitations. First, some of the existing citation recommendation systems require the entire article [2], [3] (in some cases, including a set of seed citations [4], [5]) to be submitted as input. Further, some other existing methods [2], [3] need the citation context as input to suggest the appropriate references for the given article. This implies the assumption that the person conducting the search is confident of the novelty of contributions in the article (which is why he/she chooses to invest sufficient time to create the article). This reduces the usefulness of such citation recommendation systems to only as a refinement tool, and not a tool which can potentially be used at the beginning of the acedamic research. Also, since obtaining the latent information such as citation context is costly, finding the recommendation for citations given a set of keywords is extremely useful for many scientific purposes. Second, most of the citation recommendation systems aim at suggesting prestigious and well-cited articles; they do not address the need to provide a diverse set of citations, based on a keyword driven search for a particular topic. However, we often need recommendation of diversified citations of the given query topic; for instance, it helps authors to write survey papers on a topic and it helps scholars to get a broad view of key problems on a topic. While there has been work on diversifying the citation recommendations [5] that takes a seed set of citations as input, such a system does not show a human correlatable performance improvement (i.e., they show that the results are more diverse, but it is not clear why the more diverse results are more valuable). Third, one of the problems in the keyword based citation recommendation is that the search results typically

978-1-4799-7964-6/15/$31.00 © 2015 IEEE



(a) A hypothetical network

(b) Ranking based on PageRank

(c) Ranking based on DiSCern

Fig. 1. (a) A stylized citation network, (b) Top three nodes found using PageRank are highlighted with large circles, and (c) Top three nodes found using DiSCern are highlighted with large circles.

would not include the semantically correlated articles if these articles do not use exactly the same keywords.

To the best of our knowledge, there is no known citation recommendation system in the literature that addresses the above three limitations simultaneously. In this paper, we propose a novel diversified citation recommendation system called **DiSCern** *to precisely address the above research gap.* Our proposed approach builds on a reinforced random walk [6], [7] on the citation network and it does not utilize any latent information of the articles. Our proposed approach also preserves the direction awareness property, which allows to give more importance to either the citation of papers or their references; it also makes the citation suggestion process easily tunable for finding either recent or traditional relevant papers [8].

The key ingredients of our proposed citation recommendation system, DiSCern, are as follows: (i) Given a scientific query, our system first expands the query in order to cover as much space of related topics as possible by leveraging a keyword-keyword graph, which is constructed using the co-occurrence of keywords in the articles; (ii) the articles corresponding to the expanded query are collected to determine an induced subgraph from the original citation network; and (iii) we run an efficient reinforced random walk based algorithm on this induced subgraph to come up with the citation recommendations.

Since most of the existing citation recommendation systems use either PageRank [9] or its variants [10] to rank the articles [2], [8], we consider PageRank based ranking of the articles as the standard baseline algorithm while evaluating the performance of our proposed approach. Figure 1 illustrates the ranking outcomes of the PageRank and the proposed DiSCern algorithms on a stylized citation network. For the citation network shown in Figure 1(a), the PageRank algorithm selects the top three nodes based on their prestige in the network as highlighted in Figure 1(b). It can be seen that these top three nodes are clustered. However, DiSCern considers both prestige and diversity; it finds the top three nodes as highlighted in Figure 1(c). Clearly, these top three nodes found using DiSCern cover the entire network.

A. Contributions of the Paper

The primary contribution of this paper is to propose a novel diversified citation recommendation system (*DiSCern*), which takes keyword(s) as the topic(s) of search. We now briefly summarize the major contributions of this paper as follows:

- Contrary to the existing methods where citation contexts [2] or bibliographic information [8] serve as the seeds of the citation recommendation system, *our proposed system takes only keyword(s) as the topic(s) of search*;
- The adaptation of a time-variant random walk process, also known as vertex-reinforced random walk [6], [7] takes into account both prestige and diversity in the citation recommendation task.
- The performance of DiSCern is significantly enhanced with the inclusion of a novel query expansion step based on communities formed by the keywords in a keyword-keyword graph. The communities are found as a preprocessing step, and this results in making the final recommendation faster as shown in the experimental results.
- The experiments conducted in this paper are based on a large dataset of more than 1.7 million scientific papers of computer science domain. We are also planning to make this dataset publicly available for the research purposes. We also run our system on a dataset of more than $29,000$ papers in theoretical high-energy physics domain to further confirm our findings.
- We compare the performance of DiSCern with that of the PageRank based algorithms and *we obtain a significant improvement in the quality of citation recommendations in terms of various relevance and diversity metrics.*

B. Organization of the Paper

The rest of the paper is organized as follows. In Section II, we discuss the related work in the fields of scientific paper and citation recommendation systems. In Section III, we present a brief overview of the random walk based ranking on which our baseline system is constructed. We then introduce our system and discuss its theoretical foundation and computational approximations in Section IV. Next, in Section V we provide the experimental results which comprises of a description of the datasets and a detailed analysis of the results. We conclude the paper in Section VI with pointers to future work.

Note: We use the two terms *graph* and *network* interchangeably throughout this paper.

II. Related work

Citation recommendation system: One can categorize the approaches proposed in the literature based on several dimensions. One dimension is the specification of query, i.e., what do the user provide as the input query to find the citations. Another dimension is the type of information needed for the candidate citations. Some approaches require a set of seed citations to be given as query [4], [5], [8]. These approaches require only the citation network of the candidate articles. Other approaches [2], [3], [11], [12] required the users to provide the entire article text as the input. These approaches also require significant latent information (the full text of candidate articles). In fact, [12] requires both the full article and the citation graph of the candidate articles. [11] generates queries using terms in the paper, finds candidates for recommendation from the web and ranks the candidates based on the content of abstract and title.

While [2] required the authors to identify the places, in a full manuscript, where they wanted the citations to be recommended for (known as citation context), in [3] the appropriate citation contexts also were determined automatically by comparing against the previous research articles. These methods only took care of local neighbors (i.e., citations and references) into account, e.g., bibliographic coupling [13], co-citation [14] and CCIDF [15]. In [12], one takes the query article and finds similar articles from the candidate citation database. This set is enhanced with all the articles that were referred to by the candidate articles that are similar to the query article. Now, a set of features, including Katz distance [16] are used in the task of identifying the citations in a classification approach.

Random walk with restarts [17] and well-known PageRank [9] are used by PaperRank [4] and ArticleRank [18] approaches to rank the citations using the citation graph. A seed set of citations is used to bias the ranking (for relevance) using a personalized page rank [10] method in PaperRank [4] for using these rankings for recommending the citations. [19] proposed to use co-access digital records for papers in case of explicit citation network not available, specially for new papers who have not yet acquired enough citations.

A few hybrid recommendation systems have been built on the basic of both topic and link mining. Shaparenko and Joachims [1] proposed a technique based on language modeling and convex optimization to recommend documents. Torres et al. [20] used a combination of context-based and collaborative filtering algorithms to build a recommendation system, and reported that the hybrid algorithms performed better than individual ones. Caragea et al. [21] showed that singular value decomposition based approach outperforms collaborative filtering.

Result diversification. The importance of diversity in ranking has been discussed in various data mining fields, including text retrieval [22], recommender systems [23], online shopping [24], and web search [25]. Although there is no single definition of diversity, different objective functions and axioms that are expected to be satisfied by a diversification system were discussed in [26]. Diversification of the results of random walk-based methods on graphs only attracted attention recently. One of the earlier algorithms that address diversified ranking on graphs by vertex selection with absorbing random walks is GRASSHOPPER [27]. The shortcomings of this technique are discussed in [28] in detail. Recently, a detailed survey is conducted [5] on the problem of result diversification in citation-based bibliographic search. This paper takes a set of seed citations and finds a more complete and diversified set of citations using the Vertex Reinforced Random Walk. While it shows that the resulting set of citation is more diverse, this paper does not demonstrate that the diversification results in any observable improvement from the researcher's perspective (i.e., better precision or recall).

The basic difference of our approach from the other methods is that we use only the keywords associated with the articles and the topological structure of the underlying citation network for ranking and recommending citations. We believe that additional meta data information such as abstract, citation contexts, etc. might enhance the system; but accumulating such meta data is challenging and time-consuming for most of the scientific domains. Moreover, the graph-based recommendation system would generate a robust and generic system that can be substantiated for recommending other items as well, such as academic collaborators, movies, online items etc.

III. Ranking with Random Walk

Here we present a brief overview of random walk process for ranking purposes.

Preliminaries: Let $G = (V, E)$ be a graph where V is a finite set of vertices and E is a finite set of edges. We define an ordered pair (u, v) as an edge from vertex u to vertex v. When G is an undirected graph, we have $(u, v) = (v, u)$; when G is a directed graph, (u, v) and (v, u) are different. We define the weight of an edge using $w(u, v)$. Note that, $w(u, v)$ would take either a binary value or any real value. Then the task of ranking the vertices can be casted as the prestige in a network by finding a suitable prestige function $f : V \rightarrow \mathbb{R}+$. Beyond simple measures such as degree for estimating prestige, recent research focuses on a family of centrality measures based on the stationary distribution of a random walk in the network, such as the well-known PageRank [9] or its alternatives like LexRank [29] etc.

Formulation: A family of prestige measures in networks leverages the stationary distribution of a random walk in the network. A random walk defines a Markov chain in the given (either directed or undirected) network, where each vertex represents a state and a walk transits from one state to another based on a transition probability, denoted as $p(u, v)$. In other words, a random walk on G is defined by a transition probability function $p : V \times V \rightarrow [0, 1]$. Let us use $p_T(u)$ to denote the probability that the walk is at state u at time T. A standard random walk can be defined as

$$p_T(v) = \sum_{(u,v)\in E} p(u,v) p_{T-1}(u) \tag{1}$$

If the Markov chain is ergodic, $p_T(v)$ converges to a stationary distribution $\pi(v)$ which is commonly used to measure the importance of vertices. Most existing random walk models assume that the transition probability $p(u,v)$ does not change over time, and instead can be estimated based on the topological structure of the network and/or the prior knowledge about the process. However, there are different ways to estimate $p(u,v)$. For instance, in a web hyperlink graph, $p(u,v)$ can be estimated by

$$p(u,v) = \begin{cases} (1-d)\frac{1}{N} + d \cdot \frac{I[(u,v)\in E]}{deg(u)}, & \textit{if } deg(u) > 0 \\ \frac{1}{N}, & \textit{otherwise} \end{cases} \tag{2}$$

where d is a damping factor, $deg(u)$ is the out-degree of the web page u (the number of hyperlinks from u to other web pages) and $I(X)$ is an indicator function which returns 1 if the expression X is true, and zero otherwise. The Markov chain defined by $p(u,v)$ is ergodic. The stationary distribution of this random walk, $\pi(u)$, yields to the well known PageRank score for ranking web pages.

More generally, for a given weighted graph, one can estimate $p(u,v)$ by substituting $w(u,v)/\sum_{v\in V} w(u,v)$ for $I[(u,v) \in E]/deg(u)$ in Equation (2). In another scenario where we have a prior distribution $p^*(v)$ (s.t. $\sum_v p^*(v) = 1$), we can substitute $1/N$ in Equation (2) with $p^*(v)$. The stationary distribution of such a random walk then yields to personalized PageRank, or topic-sensitive PageRank [10].

In all these cases, we notice that the transition probabilities do not change throughout the random walk process. In other words, the corresponding Markov chain is time-homogeneous. π assigns higher weights to vertices that are more prestigious. If one vertex is visited very frequently by the walk, all its neighbors are also more likely to be visited, thus inherit a prestige score from that vertex. This is known as a *regularization process* [30], [31], or a smoothing process [32] of scores in the network. In the scenario that vertices with high degrees are well connected, the top ranked vertices are likely to be clustered. In other words, the top ranked results do not cover the entire network and thus are not diverse (for example, as shown in Figure 1).

Now the important question is *how to achieve diversity in a random walk?* We may expect that there is not only a smoothing process between neighbors, but also a competing process. By doing this, we expect that rich nodes get richer over time and "absorb" the scores of its neighbors. In the next section, we propose a principled way to facilitate this mechanism.

IV. Our Proposed Approach

In this section, we first present the proposed ranking method and then present our algorithms.

A. Proposed Ranking Method: DiSCern

Our proposed model primarily builds on a time-variant random walk process, known as the *vertex-reinforced random walk* (*VRRW*) [6], [7], which contrary to the general *PageRank* algorithm [9], takes into account both *prestige* and *diversity* in order to rank the vertices in a network. A brief description of the mathematical formulation of vertex-reinforced random walk is presented below.

1) Vertex-Reinforced Random Walk (VRRW): Most of the time-homogeneous random walk processes such as PageRank assume that the transition probabilities of edges (i.e, the probability of jumping from one node to another through their interconnection) always remain constant over time. However, in a real-world scenario, one can reasonably consider the transition probabilities as a factor of time. For instance, a visitor is more likely to favor that place which has already been recognized by many other visitors; an actor accumulates prestige when acting in various movies, and the prestige in turn helps her get even more opportunities. These can all be considered as various random walk processes with varying transition probabilities. The underlying formulation of vertex-reinforced random walk process is motivated by this idea – *the transition probability to one vertex from others is reinforced by the number of previous visits to that vertex.* In the rest of the section, we briefly discuss our model with the notations introduced as in Section III.

Formulation. Formally, let $p_0(u,v)$ be the transition probability prior to any reinforcement and let $N_T(v)$ be the number of times the walker has visited v up to time T. Then a VRRW can be defined sequentially as follows. First, we initialize $N_0(v) = 1$ for $v = 1, ..., n$. Suppose, we know the random walker stays at state u at time T, then at time $T+1$, the random walker moves to state v ($v = 1, ..., n$) with probability $p_T(u,v) \propto p_0(u,v)N_T(v)$ for any state u. In other words, $p_T(u,v)$ is reinforced by $N_T(v)$. Pemantle [6] showed that under some well-defined conditions, the score in VRRW almost surely converges to some stationary distribution.

We now introduce the general form of our model, $DiSCern$, based on a similar reinforced random walk. Let $p_T(u,v)$ be the transition probability from any state u to any state v at time T. Then a family of time-variant random walk processes can be defined in which $p_T(u,v)$ satisfies the following equation [6]:

$$p_T(u,v) = (1-\lambda)\cdot p^*(v) + \lambda \cdot \frac{p_0(u,v)\cdot N_T(v)}{D_T(u)} \tag{3}$$

where

$$D_T(u) = \sum_{v\in V} p_0(u,v) N_T(v) \tag{4}$$

Here, $p^*(v)$ is a distribution which represents the prior preference of visiting vertex v. When $p^*(v)$ is uniform, the left component of Equation (3) is similar to the random jumping probabilities in PageRank. $p^*(v)$ could also be realized as a topic-sensitive distribution, similar to the personalized jumping probability in personalized PageRank [10]. $p^*(v)$ could even be realized as the stationary distribution of a time-homogeneous random walk (e.g., PageRank).

$p_0(u, v)$ is the "organic" transition probability prior to any reinforcement, which can be estimated as in a regular time-homogeneous random walk. After each step, the transition probabilities will be reinforced by the expected number of visits to each vertex. It is reasonable to assume that at any time, there is a probability that the walk stays at the current state, and this probability is reinforced by the number of visits at the current state. In other words, we assume there is always an "organic" link from a vertex to itself. We have

$$p_0(u, v) = \begin{cases} \alpha \cdot \frac{w(u,v)}{deg(u)}, & \text{if } u \neq v \\ 1 - \alpha, & \text{if } u = v \end{cases}$$

where the parameter α controls the strength of self-links [6].

If the network is ergodic, after a sufficiently large T, the reinforced random walk defined by Equation 3 also converges to a stationary distribution $\pi(v)$ [7], i.e.,

$$\pi(v) = \sum_{u \in V} p_t(u, v)\pi(u), \ \forall t \geq T \tag{5}$$

$\pi(v)$ is then used to rank the vertices in the network. Apparently, $\sum_{v \in V} \pi(v) = 1$.

Suitable approximations. In the first part of this section, we introduced the general form of DiSCern based on a general reinforced random walk. Note that, the expectation of $N_T(v)$ follows the recurrent formula mentioned below [7]:

$$E[N_{T+1}(v)] = E[N_T(v)] + p_{T+1}(v) \tag{6}$$

where $p_{T+1}(v) = \sum_u p_T(u, v)p_T(u)$. It is showed that if $\pi(v)$ exists, we have $E[N_T(v)] \propto \pi(v)$, when T is sufficiently large [6]. However, in our model $p_T(u, v)$ depends on $N_T(v)$ and tracking $N_T(v)$ is non-trivial. Therefore, an efficient approximation is needed for practical applications.

In the original study of vertex-reinforced random walk, Pemantle [6] proposed an approximation as follows: Let $1 \ll L \ll T$, we can assume that the random walk process from time T to $T + L$ behaves as if $N_{T+L}(v)$ does not change over $N_T(v)$ since $L \ll T$. Therefore, the random walk in this period approximates a time-homogeneous Markov chain with a fixed transition probability $p_T(u, v)$. Since $L \gg 1$, we may also assume that $N_{T+L}(v) - N_T(v)$ is proportional to the stationary distribution of such a Markov chain, $\pi_T(v)$. We can thus approximate $N_{T+L}(v)$ using $N_T(v) + L \cdot \pi_T(v)$.

This approximation, however, is still computationally inefficient. To find $\pi(v)$, one needs to compute the stationary distribution, $\pi_T(v)$, of many different Markovian random walks. In this section, we propose a practical approximation of DiSCern.

2) Pointwise DiSCern: One way to simplify the computation is to use $p_t(v)$ directly to approximate $E[N_t(v)]$. Indeed, when the random walk reaches the stationary status (at time T), $p_t(v)$ converges to $\pi(v)$. When the walk continues running for a sufficiently long time ($t \gg T$), $E[N_t(v)]$ is proportional to $\pi(v)$, or $p_t(v)$. With this simple approximation, we have

$$E[N_T(v)] \propto p_T(v) \tag{7}$$

We denote this simple approximation as $pointwise\ DiSCern$. Equation 3 is then be simplified as

$$p_T(u, v) = (1 - \lambda) \cdot p^*(v) + \lambda \cdot \frac{p_0(u, v) \cdot p_T(v)}{D_T(u)} \tag{8}$$

where $D_T(u) = \sum_{v \in V} p_0(u, v)p_T(v)$. In the rest of our paper, we use $DiSCern$ to refer to this *pointwise DiSCern* unless otherwise stated.

B. Proposed Algorithms

Here we first describe the key ingredients of our proposed citation recommendation system and then we present the algorithms.

1) Citation Network Construction: Since our method is primarily based on a network, we construct a paper-paper citation network as follows. A citation network is defined as a graph $G =< V, E >$ where each node $v_i \in V$ represents a paper and a directed edge e_{ji} pointing from v_j to v_i indicates that the paper representing v_j cites the paper representing v_i in its references. We also add a self-link to each node.

2) Keyword Network Construction: Since the preliminary idea of our system is to recommend diversified citations for a particular search query, we intend to expand the input query to obtain a set of similar keywords that can cover different semantics of the query. For this, we use the keyword meta data available with the articles. We then construct the keyword-keyword graph as follows: We construct the keyword-keyword graph as an undirected and weighted graph $G_k(V_k, E_k)$ where each node in V_k represents a keyword and two nodes v_i^k and v_j^k ($v_i^k, v_j^k \in V_k$) are connected by an edge e_{ij}^k ($\in E_k$) if there is at least one article that contains both the keywords corresponding to these two vertices. The weight $w_{i,j}^k$ associated with an edge e_{ij}^k is determined by the number of articles where both the keywords corresponding to v_i^k and v_j^k appear.

3) Query Expansion by Clustering Keywords: Our next task is to cluster similar keywords from the topological structure of the keyword-keyword network. In the complex network analysis, clusters are often termed as *communities* where nodes with similar structural or functional properties are grouped together [33]. We utilize *Louvain* [34], a well known state-of-the-art algorithm to find communities in the keyword-keyword graph. The algorithm uses a greedy optimization method that attempts to optimize a goodness measure, named "modularity" [35] of a partition of the network. The optimization is performed in two steps. First, the method looks for "small" communities by optimizing modularity locally. Second, it aggregates nodes belonging to the same community and builds a new network whose nodes are the communities. These steps are repeated iteratively until a maximum of modularity is attained and a hierarchy of communities is produced. Although the exact computational complexity of the method is not known, the method seems to run in time $O(nlogn)$ with most of the computational effort spent on the optimization at the first level.

Now given an input query, the system first identifies the community membership of this query and then all the constituent keywords present in that community are fetched for the next step of the framework. We refer to the query expansion step as **QExpn**.

4) Retrieving Diverse and Relevant Citations: After expanding the given input query, we obtain a *expanded query* containing a set of similar keywords using QExpn described as above. Then the articles corresponding to the expanded query are collected to further determine an induced subgraph from the original citation network. We now run DiSCern on this induced subgraph to come up with the citation recommendations. We refer to this as *LocDiSCern* since we use DiSCern along with QExpn. Algorithm 1 presents the steps in LocDiSCern. Note that, LocDiSCern is convenient enough to accept multiple query keywords as well. In that case, we map each of the input query keyword to its corresponding community in the keyword-keyword graph and collect the articles of all the mapped communities, which in turn form the expanded query of the keywords.

Algorithm 1: LocDiSCern

```
Input: Citation network $G = (V, E)$, keyword-keyword
       network $G_k = (V_k, E_k)$ and a query keyword
Output: Top $K$ diverse as well as relevant citation
        recommendations
1 Given the query keyword, apply *QExpn* procedure
  wherein we expand given keyword with relevant
  keywords using $G_k$
2 Collect the set of all the articles that contain at least one
  keyword from the expanded query and call this set $S$
3 Construct the induced subgraph of $S$ from the citation
  network $G$ and we refer this $G_S$
4 Run DiSCern on $G_S$ to get relevance scores
5 Rank the nodes (or articles) of $G_S$ in non-increasing
  order of their scores
6 Return top $K$ articles from this rank list
```

Note that one might argue on the use of the keyword expansion step in LocDiSCern. However, we can also run DiSCern directly on the entire citation network by omitting the QExpn step. We refer to this as *GloDiSCern* and Algorithm 2 presents the key steps of the same.

Algorithm 2: GloDiSCern

```
Input: Citation network $G = (V, E)$ and a query
       keyword
Output: Top $K$ diverse as well as relevant citation
        recommendations
1 Run DiSCern on the citation network $G$ to get the
  relevance scores for all the articles
2 Collect the set of articles associated with the given
  query keyword and call this set $T$
3 Rank the articles in $T$ in non-increasing order of their
  scores
4 Return top $K$ articles from this rank list
```

Note that, other alternatives can also be possible wherein DiSCern is applied on the entire graph first, followed by QExpn and then the top K results can be returned. We have also attempted this approach, but did not get significantly good results.

C. System Overview

Figure 2 shows a schematic diagram of our proposed model *LocDiSCern*. The framework is mostly divided into three layers. At the top layer, users can input one or multiple queries separated by some delimiter. The input query set is then processed further for tokenization and the query tokens are individually identified from the query set. Note that, query processing [36] and query segmentation [37] are themselves two fundamental research topics in Natural Language Processing and Information Retrieval, and therefore are beyond the scope of the present study. Here, we assume that users choose the queries from the keyword pull present in our dataset and multiple queries (if any) are separated by a predefined delimiter. After query tokenization, each individual query is mapped to one of the communities obtained from the keyword network and constituent members of that community are retrieved into an extended query set. Following this, the papers associated with individual keywords in the extended query set are collected, and an induced subgraph is constructed using those papers. Next, we run the DiSCern algorithm on the induced subgraph and obtain the scores of the vertices. The vertices are then ranked in non-increasing order of their scores and top K papers are then shown to the users.

V. Experimental Results

In this section, we first describe the datasets used in our experiments. Then we highlight the baseline algorithms using which we compare and contrast the proposed algorithms. We next present the experimental results.

A. Datasets

1) Computer Science Publication Dataset: In the literature, most of the experiments on recommending citations have worked with small datasets. However in our experiments, we gather and analyze a large dataset to evaluate the performance of the proposed algorithms. We have crawled one of the largest citation datasets from Microsoft Academic Search (MAS)[1] which houses over 4.1 million publications and 2.7 million authors with updates added every week. We collected all the papers published in the computer science domain and indexed by MAS[2]. The crawled dataset contains more than 2 million distinct papers altogether which are further distributed over 24 fields[3] of computer science domain (see Table II). Moreover, each paper comes along with various bibliographic information – the title of the paper, a unique index for the paper, its author(s), the affiliation of the author(s), the year of publication, the publication venue, the related field(s) of the paper. Each paper is also annotated by MAS with a set of keywords to characterize the paper. Total 37,089 unique keywords are present in this dataset.

Preprocessing of the crawled dataset: The crawled data had several inconsistencies that were removed through a series of steps. We also removed few forward citations which point to the papers published after the publication of the source paper.

[1] academic.research.microsoft.com

[2] The crawling process took six weeks and was completed in January, 2014.

[3] A field is a sub-area of a research domain. For instance, Algorithm and AI are the two examples of fields in computer science domain.

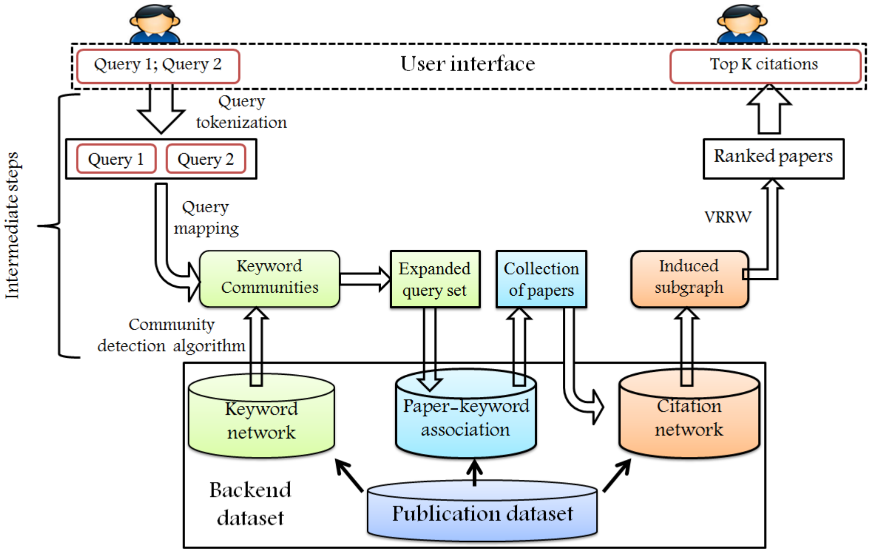


Fig. 2. A schematic diagram of our proposed model, *LocDiSCern*.

TABLE I. GENERAL INFORMATION OF RAW AND FILTERED DATASET.

	Raw	Filtered
Number of valid entries	2,473,171	1,763,837
Number of links	18,674,361	14,533,196
Number of entries with no venue	343,090	–
Number of entries with no author	45,551	–
Number of entries with no publication year	191,864	–
Partial data of the years before 1970 and 2011-2012	343,349	–
Number of authors	1,186,412	821,633
Avg. number of papers per author	5.18	5.04
Avg. number of authors per paper	2.49	2.67
Number of unique venues	6,143	5,938
Percentage of entries with multiple fields	9.08%	8.68%

TABLE II. PERCENTAGE OF PAPERS IN VARIOUS FIELDS OF COMPUTER SCIENCE DOMAIN.

Fields	**% of papers**	**Fields**	**% of papers**
AI	12.64	Algorithm	9.89
Networking	9.41	Databases	5.18
Distributed Systems	4.66	Comp. Architecture	6.31
Software Engg.	6.26	Machine Learning	5.00
Scientific Computing	5.73	Bioinformatics	2.02
HCI	2.88	Multimedia	3.27
Graphics	2.20	Computer Vision	2.59
Data Mining	2.47	Programming Language	2.64
Security	2.25	Information Retrieval	1.96
NLP	5.91	World Wide Web	1.34
Education	1.45	Operating Systems	0.90
Embedded Systems	1.98	Simulation	1.04

These forward citations are present because these are few papers that are initially uploaded in public repositories (such as http://arxiv.org/) but accepted later in a publication venue. Further, we considered only those papers published in between 1970 and 2010 and that cite or are cited by at least one paper (i.e., we removed disconnected nodes with zero in-degree or zero out-degree). In the filtered dataset, 8.68% papers belong to multiple fields (such as interdisciplinary papers). Some of the authors had been found missing in the information of the corresponding papers which were found by the DOI (Digital Object Identifier) of the publications. We also thoroughly checked the filtered papers (around 1.7 million in number) with the author and metadata information from DOI and kept only the consistent ones. Some of the references that pointed to such papers absent in our dataset (i.e., dangling references) had also been removed and the dataset was constructed accordingly to keep it as close as possible to the real citation network. The general information pertaining to the filtered dataset is presented in Table I. *We plan to make this dataset publicly available soon so that other researchers can also use it for their work.*

2) High-Energy Physics Dataset: This dataset presents information on papers in theoretical high-energy physics. This dataset is derived from the abstract and citation files provided for the 2003 KDD Cup competition[4]. Along with the citation information of each paper, it also contains the metadata information of each paper such as the name of the authors, the year of publication, the name of the journal where it is published, the subfield and the abstract. Table III provides details of this dataset.

TABLE III. GENERAL INFORMATION OF HIGH-ENERGY PHYSICS DATASET.

Number of valid entries	29,555
Number of links	352,807
Number of authors	8,987
Number of unique venues	697

Since the keyword information is missing in the dataset, we mine the keywords from the abstract of each paper. We use "Kea"[5], a keyword extraction tool from the text documents [38]. Kea identifies candidate keyphrases using lexical methods, calculates feature values for each candidate, and uses a machine-learning algorithm to predict which candidates are good keyphrases. The machine learning scheme first builds a prediction model using training documents with known keyphrases, and then uses the model to find keyphrases in new documents. We find 418 unique keywords in this dataset.

[4] http://www.cs.cornell.edu/projects/kddcup/datasets.html

[5] http://www.nzdl.org/Kea/

B. Baseline Algorithms

In this paper, we use PageRank as a baseline method to compare with our proposed method. There are two reasons behind adopting the PageRank as a baseline system: (i) Majority of the existing graph-based citation recommendation systems have been built on top of the PageRank algorithm and are proven to be very effective. Since our proposed method DiSCern is a graph-based citation recommendation system, we decided to compare DiSCern with PageRank. (ii) PageRank and DiSCern both rely only on the underlying citation network, while most other existing citation recommendation systems are heavily dependent on the entire content of each paper for recommending the citations. Such systems take the citation context or the entire contents of the article as input. Therefore, from the stand point of fair comparison, we have not considered the above kind of citation recommendations as baselines.

In fact, gathering the entire content of the papers is also time consuming and might not be feasible for all research domains. Hence, in our proposed graph-based recommendation system approach, we work with citation network of the articles. Constructing the citation network of scientific articles is an easy task (using the bibliographic information) compared to gathering the contents of the articles.

For the comparative analysis, on similar lines as in Algorithm 1 and Algorithm 2, we define *LocPageRank* and *GloPageRank*. Algorithm 3 and Algorithm 4 describe the key steps of LocPageRank and GloPageRank respectively.

Algorithm 3: LocPageRank

```
Input: Citation network $G = (V, E)$, keyword-keyword
       network $G_k = (V_k, E_k)$ and a query keyword
Output: Top $K$ diverse as well as relevant citation
        recommendations
1 Given the query keyword, apply QExpn procedure
  wherein we expand given keyword with relevant
  keywords using $G_k$
2 Collect the set of all the articles that contain at least one
  keyword from the expanded query and call this set $S$
3 Construct the induced subgraph of $S$ from the citation
  network $G$ and we refer this $G_S$
4 Run PageRank on $G_S$ to get relevance scores
5 Rank the nodes (or articles) of $G_S$ in non-increasing
  order of their scores
6 Return top $K$ articles from this rank list
```

Note that the additional step of query expansion serves as one of the key contributions of this paper. Therefore, only *GloPageRank* remains as an actual baseline system. However, we intend to see the effect of query expansion in the baseline system as well and therefore study the results of *LocPageRank*.

C. Preparing Ground-Truth (or Gold-Standard) Dataset

Essentially, one of the goals of this paper attempts to achieve is that the researchers would get rid of the frantic exercise of choosing references while writing articles like *review* or *survey articles*[6]. Therefore for evaluating our system, we manually collected a set of survey papers from both the datasets. We searched for the keywords such as "literature", "survey", "review" in the title of the papers. In computer science dataset, we made it sure that the selected papers cover all the subareas (see Table II) of computer science domain and all ranges of citations (both highly- and low-cited papers) and selected 100 such papers. In physics dataset, only 58 papers contained at least one of the above keywords in the title and we selected all of them. The keywords associated with the collected papers then form the test suites for the evaluation. Now for each paper in the gold-standard, we know the references that the current paper has cited. We assume that these references are diverse and serve as the gold-standard in our experiment. However, another problem is that all the references of a paper p present in our test-suite are published on or before the publication of p. But our system might also return references from recent published papers for the corresponding query, which may degrade the final performance of the system. Therefore for the evaluation purpose, we design a customized version of *LocDiSCern* for the keyword set present in each paper p in the gold-standard set – after keyword expansion step mentioned in Section IV-B when we collect the papers associated with these keywords, we only retain those papers published on or before the publication of p and construct the induced subgraph out of these papers. This would indeed indicate how good our system is in predicting those references used in paper p. It is worth noting that, while measuring the accuracy for the query associated with paper p, p itself is removed from the dataset to avoid any unnecessary bias. However, in the result section, we also evaluate our model in terms of the publication year of the recommend papers just to give an intuitive evidence that how our system is able to cover the diverse range of citations from both older and recent years. In that case, we omit the customization technique mentioned above.

Algorithm 4: GloPageRank

```
Input: Citation network $G = (V, E)$ and a query
       keyword
Output: Top $K$ diverse as well as relevant citation
        recommendations
1 Run PageRank on the citation network $G$ to get the
  relevance scores for all the articles
2 Collect the set of articles associated with the given
  query keyword and call this set $T$
3 Rank the articles in $T$ in non-increasing order of their
  scores
4 Return top $K$ articles from this rank list
```

D. Evaluation Metrics

We evaluate the quality of the citation recommendations suggested by both the proposed and the baseline algorithms with a number of metrics [5], [8]. We describe a few well known metrics for relevance and diversity as well as certain other metrics for this purpose.

[6] A review/survey article is assumed to cover a vast area of research from diverse perspective on which the article is based.

1) Relevancy Metrics: For relevance, we use the simple IR based metric as follow:

(a) Recall ($R@K$): The recall is defined as the percentage of original citations for a query (corresponding to each paper p in the gold-standard set) that appear in the top K recommended citations. It essentially indicates how effective is the system to predict the references used while the paper p was written.

(b) Mean Average Precision ($MAP@K$): It is mostly used in ranked-based retrieval. Since we also wish to rank the most relevant and diverse citations at the top, we adopt this metric to check how accurate is our ranking. Given each query q, let the top K citations returned by the system are $c_1, c_2, ..., c_K$ in the ranked order of relevance. Then MAP corresponding to q is defined as

$$MAP_q = \frac{\sum_{i=1}^{K} Precision(c_1, c_2, ..., c_i)}{K} \tag{9}$$

where $Precision(c_1, c_2, ..., c_i)$ is the fraction of relevant references among $c_1, c_2, ..., c_i$. Then for the query set Q, the overall MAP is defined as $MAP@K = \frac{\sum_{q \in Q} MAP_q}{|Q|}$.

(c) Co-cited probability ($CP@K$): We may recommend some relevant or even better recommendations other than those original ones among the top K results, which cannot be captured by the traditional metric like precision. The previous work usually conducted user studies for this kind of relevance evaluation [39], [40]. In this paper, we instead use the wisdom of the population as the ground-truth to define a systematic metric [2]. For each pair of documents $< d_i, d_j >$ where d_i is an original citation and d_j is a recommended one, we calculate the probability that these two documents have been co-cited by the popularity in the past as

$$CP = \frac{number\ of\ papers\ citing\ both\ d_i\ and\ d_j}{number\ of\ papers\ citing\ d_i\ or\ d_j} \tag{10}$$

For each query, the co-cited probability is then averaged over all $K \cdot l$ unique document pairs for the top K results, where l is the number of original citations. Then $CP@K$ is obtained by averaging all CPs by the number of queries.

2) Diversity Metrics: Our preliminary assumption while building the gold-standard dataset (Section V-C) was that the survey papers usually contain diverse references. Therefore, we believe that both $R@k$ and $MAP@K$ convey the diversity of the results along with the relevance. However, we also adopt standard graph-based metrics used previously in recommendation systems [5], [8] to evaluate the diversity of the framework.

(a) *l*-hop graph density ($den_l@K$): In our experiments, we leverage the density measure in network science. The density of a graph is defined as the ratio of the number of edges (excluding self-links) that are present in the network to the maximum possible number of edges in that network. A variant of graph density measure is the *l*-hop graph density measure [41], which takes the effect of indirect neighbors into account. It is computed as

$$den_l(S) = \frac{\sum_{u,v \in S, u \neq v} d_l(u, v)}{|S| \times (|S| - 1)} \tag{11}$$

where S is the set of top-K results, $d_l(u, v) = 1$ when v is reachable from u within l steps, i.e., $d(u, v) \leq l$; and 0 otherwise. Therefore, *l*-hop graph density is used as an inverse measure of diversity in the top-K ranked vertices. The intuition is that the smaller the value of den_l is, the more independent the top-K vertices are, thus the less redundancy and higher diversity results are contained in the top-ranked list (and vice versa). However, in order to make the higher the better and make it consistent throughput, we measure (1-den_l) for *l*-hop graph density.

(b) *l*-expansion ratio ($\sigma_l(S)$): This metric and its variance *l-expansion ration* [28] measure the coverage of the graph by the solution set. They are computed using the *l*-step expansion set given below

$$\sigma_l(S) = \bigcup_{s \in S} N_l(s) \tag{12}$$

where S is the set of top-K results and $N_l(s)$ is all the neighbors up to l hops for vertex s.

3) Other Metrics: Besides the relevance and diversity metrics, we also use other metrics to observe the difference in the results obtained using our algorithms and that of the baselines.

(a) Average publication year ($T@K$): The average publication year of the recommended set is also measured to show how diverse range of papers from different years are being recommended in the top-K set. It also indicates how the references of each gold-standard paper could have been if it is written in recent time period.

(b) Difference ratio ($DR@K$): We expect that the results of LocDiSCern is somewhat different from the top K relevant set of results returned by the other competing algorithms, as our experiments will show the set of nodes recommended by the original PageRank are not enough diverse. Therefore, we decide to measure the different of each result set of the competing algorithms from the set of top K results returned by LocDiSCern as follows:

$$DR@k = \frac{\sum_{q \in Q} DR_q}{|Q|} \tag{13}$$

where $DR_q = \frac{S \bigcap \hat{S}}{K}$; S and $\hat{S}$ are the top K results from the baseline system and our model respectively, and $|S| = |\hat{S}| = K$.

E. Experimental Setup

We first construct the citation network using each of the filtered datasets. It is possible that the resulting network can be disconnected. In that case, we work with the largest connected component. Then we construct the keyword-keyword graph, G_k and we note that the graph is extremely dense. Therefore, we filter out a few edges having low edge weights. Again there could exist several techniques to perform this filtering to obtain a sparse version of the original keyword-keyword graph. However, we use the local graph sparsification technique proposed by Satuluri *et al.* [42]. Since every node v in G_k is connected to a set of weighted spokes[7], we calculate the mean of all the weights of the spokes connected to v. Among these spokes, we then retain all those whose associated weights are greater than or equal to the mean weight corresponding to

[7] An edge $e^k_{ij} =< v^k_i, v^k_j >$ is assumed to be formed by two spokes: e^{ki}_{ij} and e^{kj}_{ij} which emit from v^k_i and v^k_j respectively.

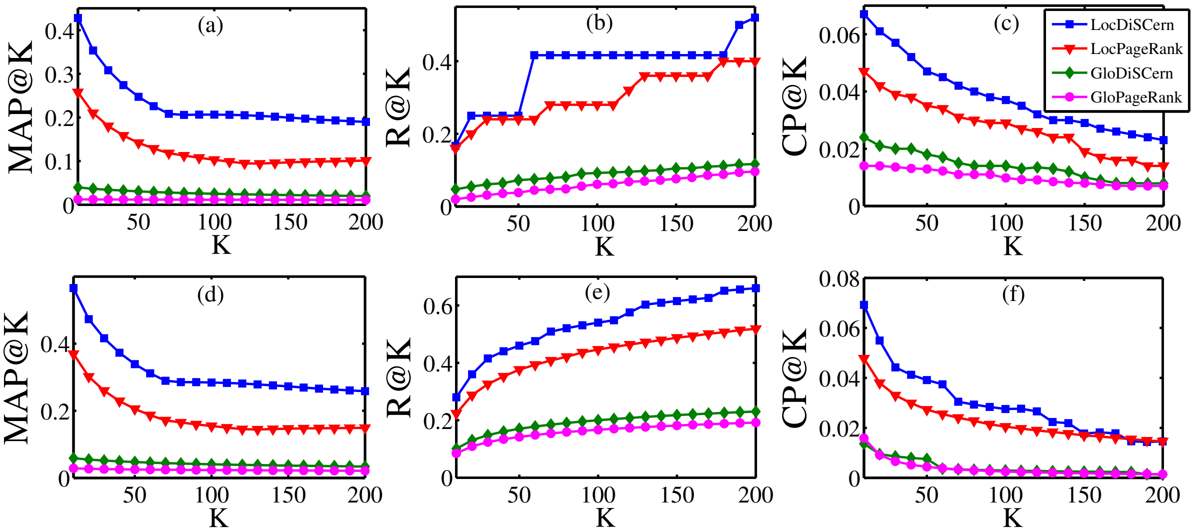


Fig. 3. (Color online) Comparison of network-based ranking algorithms in recommending citations for relevance metrics: MAP, recall and co-cited probability for computer science (top panel: (a) – (c)) and high-energy physics (bottom panel: (d) – (f)) datasets. Parameters: λ (or d) $= 0.9$ in two versions of PageRank and DiSCern; $\alpha = 0.25$ for DiSCern.

v. This method is applied to all the nodes. After the filtration, if one of the spokes associated with an edge persists in G_k, we retain that edge in the final network construction. If both the spokes associated with an edge are deleted by this process, then we remove the corresponding edge from the final network. At the end of this process, we obtain a filtered weighted graph which is more sparse than its original counterpart.

When we run Louvain [34] on the keyword-keyword graph of computer science dataset, we get 723 communities. Similarly, we get 46 communities on the keyword-keyword graph of the high-energy physics dataset.

F. *Results*

We compare the performance of the two proposed algorithms (i.e., LocDiSCern and GloDiSCern) with that of the two baseline algorithms (LocPageRank and GloPageRank). We set the jumping probability in all these methods to 0.1. Since there are separate measures for prestige and diversity, we do not tune the parameter α and simply set it 0.25. Following [27], we set the prior distribution $p^*(v)$ to be proportional to the number of inward citations that the paper (vertex) has received so far. The same information is provided to PageRank as well.

Results based on the relevance metrics:

Figure 3 shows the performance of LocDiSCern, GloDiSCern, LocPageRank, and GloPageRank with respect to the three relevance metrics ($MAP@K$, $R@K$, $CP@K$) on the computer science dataset as well as the high-energy physics dataset. Note that, for evaluating $MAP@K$ and $R@K$, we adopt the customization technique based on publication year before running the four algorithms as mentioned in Section V-C. But for the other metrics, we avoid this technique.

In Figure 3(a) and Figure 3(d), we observe that both LocDiSCern and LocPageRank outperform GloDiSCern and GloPageRank with respect to the $MAP@K$ metric. Further, the performance of LocDiSCern is more about 40% than that of LocPageRank for both the datasets. We also notice that the performance GloPageRank seems to be the lowest among the four algorithms.

In Figure 3(b) and Figure 3(e), we plot the value of recall for the four algorithms on computer science and physics datasets respectively. The observation is quite similar to that of Figure 3(a) and Figure 3(d). Surprisingly, we notice that the plots of recall for LocDiSCern and LocPageRank almost behave like a step-function for the computer science dataset. It essentially indicates that for a set of values within a certain range of K, each of them performs nearly similar. After a certain point of time, the recall value tends to increase suddenly. However, this behavior is not observed for GloDiSCern and GloPageRank. We argue that this behavior is due to the keyword expansion step. The reason could be that during the process of keyword expansion, even if a new keyword which is similar to the input query appears in the query expansion step and is not as relevant to the gold-standard article as the original query, the system still considers the papers associated with the new keyword while recommending the final results. Now if a bunch of articles corresponding to such new keywords are recommended, the recall value would tend to remain same irrespective of the increase of number of recommendations. But this may affect the precision value as shown in Figure 3(a). However, the pattern of recall for physics dataset increases almost exponentially with some minor steps observed in the plot corresponding to LocDiSCern.

In Figure 3(c) and Figure 3(f), the value of co-cited probability is plotted for the four algorithms on both the datasets. Here again, the overall observation is similar to the earlier two scenarios. We observe that the pattern of CP tends to decrease with the increase of K. It essentially indicates that even if the recommended candidates do not appear in the gold-standard dataset, they seem to be quite relevant to the input query since there exist a significant amount of

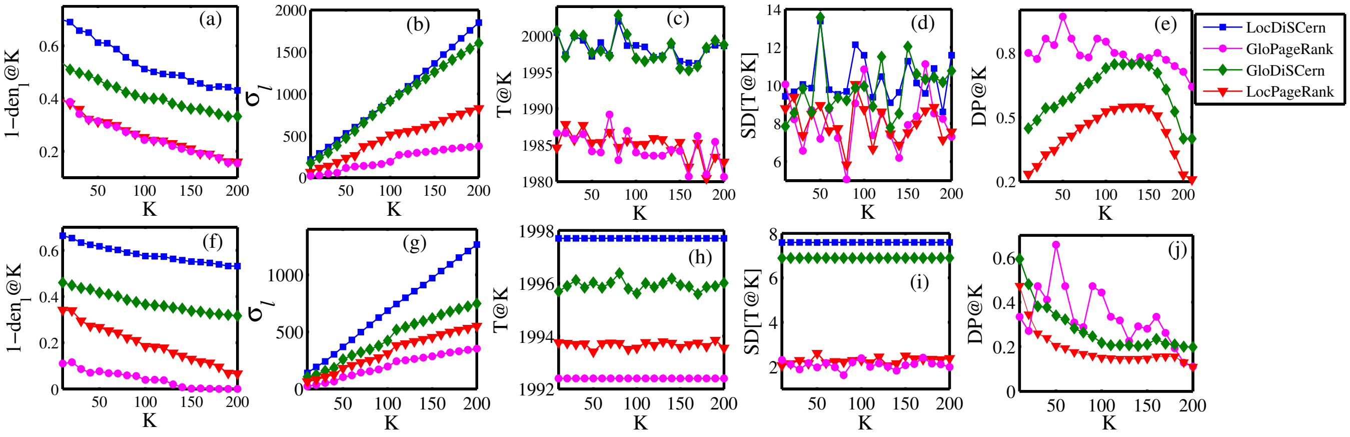


Fig. 4. (Color online) Comparison of network-based ranking algorithms in recommending citations for different diversity (l-hop graph density, l-expansion ratio) and other (average publication year, standard deviation (SD) of the publication year and the differences ratio) metrics for computer science (top panel: (a) – (e)) and high-energy physics (bottom panel: (f) – (j)) datasets. In frames (e) and (j), the differences of all the other systems are measured with respect to the LocDiSCern; therefore, no line for LocDiSCern appears in that panel. Parameters: λ (or d) $= 0.9$ in two versions of PageRank and DiSCern; $\alpha = 0.25$ for DiSCern.

papers that cite both these (recommended and gold-standard papers) simultaneously. This indeed corroborates our earlier hypothesis that our system can also recommend better citations that might not appear in the gold-standard set.

Results based on the diversity and the other metrics:

In Figures 4 (a)-(b) and (f)-(g), we present the results using the two diversity metrics. In Figure 4(a) and Figure 4(f), we observe that both LocDiSCern and GloDiSCern clearly outperform LocPageRank and GloPageRank. On similar lines, both LocDiSCern and GloDiSCern significantly outperform their counterparts using the l-step expansion ratio (see Figure 4(b) and Figure 4(g)), which is related to the coverage of the network with the recommendations. Both LocPageRank and GloPageRank perform convincingly worse with respect to these diversity metrics. In particular, the results obtained from the LocPageRank and GloPageRank are more clustered in the network compared to that of LocDiSCern as well as GloDiSCern.

In Figure 4(c) and Figure 4(h), we plot the average publication year of all the recommended candidates using each of the four algorithms on two datasets respectively. Interestingly, we notice that our proposed two DiSCern-based algorithms outperform the two baseline systems. It indicates that the vertex-reinforced random walk based methods tend to recommend mostly the recent papers as compared to the PageRank based methods. However, one might argue that the higher value of publication year might not be a good indicator to judge the time span covered by the recommended candidates. A superior citation recommendation system should recommend citations that covers a large time span, i.e., the standard deviation of the publication years of the recommended papers should be higher. In Figure 4(d) and Figure 4(i), we plot the standard deviation of the publication years of recommended papers for the four algorithms on two datasets. Here also, we observe that our approach outperforms the baselines. Therefore, we conclude that vertex-reinforced random walk based algorithms not only recommend high quality citations based on relevancy, but also they tend to recommend both older and recent citations. In Figure 4(e) and Figure 4(j), we measure the differences of the outputs obtained using the remaining three algorithms with respect to LocDiSCern. As expected, we observe that the difference is most prominent for the results obtained from GloPageRank. However, we also notice that the patterns for GloDiSCern and LocPageRank in computer science dataset are similar – an initial increase is followed by a decrease (hyperbolic shape). The reason could be that there might exist a critical value of K after which these systems tend to return almost same set of results. A good recommendation system should adopt this critical K value while recommending results for different queries. In physics dataset, the pattern is similar for GloDiSCern and LocPageRank; however it decreases exponentially with the increase of K.

VI. Conclusions and Future Work

In this paper, we present DiSCern, a novel framework that balances prestige and diversity in the task of citation recommendation. There are many potential applications of DiSCern. One can recast DiSCern to facilitate other applications such as ranking web pages in a web hyperlink graph, summarizing texts and extracting keywords and opinions from snippet generated graph. Since we have such a massive dataset with additional meta data information, an interesting future direction is to combine DiSCern with other features in a learning-to-rank framework. At present, our system only considers keywords present in our repository as inputs. In future, we would like to focus more on query processing part to make DiSCern as a full-fledged system. We also plan to release the computer science publication dataset soon for the research community to facilitate further investigations.

References

[1] B. Shaparenko and T. Joachims, "Identifying the original contribution of a document via language modeling," *Machine Learning and Knowledge Discovery in Databases*, pp. 350–365, 2009.

[2] Q. He, J. Pei, D. Kifer, P. Mitra, and L. Giles, "Context-aware citation recommendation," in *WWW*. USA: ACM, 2010, pp. 421–430.

[3] Q. He, D. Kifer, J. Pei, P. Mitra, and C. L. Giles, "Citation recommendation without author supervision," in *WSDM*, 2011, pp. 755–764.

[4] M. Gori and A. Pucci, "Research paper recommender systems: A random-walk based approach," in *Web Intelligence*, 2006, pp. 778–781.

[5] O. Küçüktunç, E. Saule, K. Kaya, and Ü. V. Çatalyürek, "Diversifying citation recommendations," *CoRR*, vol. abs/1209.5809, 2012.

[6] R. Pemantle, "Vertex reinforced random walk," *Prob. Th. and Rel. Fields*, pp. 117–136, 1992.

[7] M. Benam and P. Tarrs, "Dynamics of vertex-reinforced random walks," *The Annals of Probability*, vol. 39, no. 6, pp. 2178–2223, 11 2011.

[8] O. Kktun, E. Saule, Kamer, and mit V. atalyrek, "Direction awareness in citation recommendation," in *DBRank, in conjunction with VLDB*. Istanbul, Turkey: ACM, 2012.

[9] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web," Stanford University, Technical Report, 1998.

[10] T. H. Haveliwala, "Topic-sensitive pagerank," in *WWW*. New York, USA: ACM, 2002, pp. 517–526.

[11] C. Nascimento, A. H. Laender, A. S. da Silva, and M. A. Gonçalves, "A source independent framework for research paper recommendation," in *Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries*, ser. JCDL '11. New York, NY, USA: ACM, 2011, pp. 297–306. [Online]. Available: http://doi.acm.org/10.1145/1998076.1998132

[12] T. Strohman, W. B. Croft, and D. Jensen, "Recommending citations for academic papers," in *SIGIR*. USA: ACM, 2007, pp. 705–706.

[13] M. Kessler, "Bibliographic coupling between scientific papers." *American Documentation 14*, pp. 10–25, 1963.

[14] H. Small, "Co-citation in the scientific literature: A new measure of the relationship between two documents," *J. Am. Soc. Inf. Sci. Technol.*, vol. 24, no. 4, pp. 265–269, 1973.

[15] C. L. Giles, K. D. Bollacker, and S. Lawrence, "Citeseer: An automatic citation indexing system," in *ACM Conference on Digital Libraries*. New York, USA: ACM, 1998, pp. 89–98.

[16] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *J. Am. Soc. Inf. Sci. Technol.*, vol. 58, no. 7, pp. 1019–1031, May 2007.

[17] J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu, "Automatic multimedia cross-modal correlation discovery," in *SIGKDD*. New York, USA: ACM, 2004, pp. 653–658.

[18] J. Li and P. W. 0002, "Articlerank: a pagerank-based alternative to numbers of citations for analysing citation networks." *Aslib Proceedings*, vol. 61, no. 6, pp. 605–618, 2009.

[19] S. Pohl, F. Radlinski, and T. Joachims, "Recommending related papers based on digital library access records," in *Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries*, ser. JCDL '07. New York, NY, USA: ACM, 2007, pp. 417–418. [Online]. Available: http://doi.acm.org/10.1145/1255175.1255260

[20] R. Torres, S. M. McNee, M. Abel, J. A. Konstan, and J. Riedl, "Enhancing digital libraries with techlens+," in *JCDL*. New York, USA: ACM, 2004, pp. 228–236.

[21] C. Caragea, A. Silvescu, P. Mitra, and C. L. Giles, "Can't see the forest for the trees?: A citation recommendation system," in *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries*, ser. JCDL '13. New York, NY, USA: ACM, 2013, pp. 111–114. [Online]. Available: http://doi.acm.org/10.1145/2467696.2467743

[22] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *SIGIR*. New York, USA: ACM, 1998, pp. 335–336.

[23] C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, "Improving recommendation lists through topic diversification," in *WWW*. New York, USA: ACM, 2005, pp. 22–32.

[24] E. Vee, U. Srivastava, J. Shanmugasundaram, P. Bhat, and S. A. Yahia, "Efficient computation of diverse query results," in *ICDE*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 228–236.

[25] P. Castells, S. Vargas, and J. Wang, "Novelty and Diversity Metrics for Recommender Systems: Choice, Discovery and Relevance," in *DDR workshop, ECIR*, Dublin, 2011.

[26] S. Gollapudi and A. Sharma, "An axiomatic approach for result diversification," in *WWW*. New York, , USA: ACM, 2009, pp. 381–390.

[27] X. Zhu, A. Goldberg, J. V. Gael, and D. Andrzejewski, "Improving diversity in ranking using absorbing random walks," *HLT-NAACL*, pp. 97–104, 2007.

[28] R.-H. Li and J. X. Yu, "Scalable diversified ranking on large graphs." in *ICDM*. IEEE, 2011, pp. 1152–1157.

[29] G. Erkan and D. R. Radev, "Lexrank: Graph-based lexical centrality as salience in text summarization," *J. Artif. Int. Res.*, vol. 22, no. 1, pp. 457–479, Dec. 2004.

[30] D. Zhou, J. Huang, and B. Schölkopf, "Learning from labeled and unlabeled data on a directed graph," in *ICML*. New York, USA: ACM, 2005, pp. 1036–1043.

[31] X.-Q. Cheng, P. Du, J. Guo, X. Zhu, and Y. Chen, "Ranking on data manifold with sink points," *IEEE Trans. on Knowl. and Data Eng.*, vol. 25, no. 1, pp. 177–191, Jan. 2013.

[32] Q. Mei, D. Zhang, and C. Zhai, "A general optimization framework for smoothing language models on graph structures." in *SIGIR*. ACM, 2008, pp. 611–618.

[33] M. E. J. Newman, "Community detection and graph partitioning," *Europhys. Lett*, vol. 103, no. 1, p. 28003, 2013.

[34] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J. Stat. Mech.*, no. 10, p. P10008, Oct. 2008.

[35] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E*, vol. 69, no. 026113, 2004.

[36] M. Ubell, "The intelligent database machine (idm)," in *Query Processing in Database Systems*, 1985, pp. 237–247.

[37] M. Hagen, M. Potthast, B. Stein, and C. Bräutigam, "Query segmentation revisited," in *WWW*. New York, USA: ACM, 2011, pp. 97–106.

[38] I. H. Witten, G. W. Paynter, E. Frank, C. Gutwin, and C. G. Nevill-Manning, "Kea: Practical automatic keyphrase extraction," in *Proceedings of the Fourth ACM Conference on Digital Libraries*, ser. DL '99. New York, NY, USA: ACM, 1999, pp. 254–255. [Online]. Available: http://doi.acm.org/10.1145/313238.313437

[39] S. M. McNee, I. Albert, D. Cosley, P. Gopalkrishnan, S. K. Lam, A. M. Rashid, J. A. Konstan, and J. Riedl, "On the recommending of citations for research papers," in *CSCW*. New York, USA: ACM, 2002, pp. 116–125.

[40] K. Chandrasekaran, S. Gauch, P. Lakkaraju, and H. P. Luong, "Concept-based document recommendations for citeseer authors." in *AH*, ser. Lecture Notes in Computer Science, W. Nejdl, J. Kay, P. Pu, and E. Herder, Eds., vol. 5149. Springer, 2008, pp. 83–92.

[41] H. Tong, J. He, Z. Wen, R. Konuru, and C.-Y. Lin, "Diversified ranking on large graphs: An optimization viewpoint," in *SIGKDD*. New York, USA: ACM, 2011, pp. 1028–1036.

[42] V. Satuluri, S. Parthasarathy, and Y. Ruan, "Local graph sparsification for scalable clustering." in *SIGMOD*. ACM, 2011, pp. 721–732.