Journal of Universal Computer Science, vol. 21, no. 6 (2015), 871-889
submitted: 30/11/14, accepted: 20/2/15, appeared: 1/6/15

Similarity-based Complex Publication Network Analytics for Recommending Potential Collaborations[1]

Ngoc Tu Luong[2], Tuong Tri Nguyen, Dosam Hwang
(Department of Computer Engineering, Yeungnam University
Gyeongsan, Korea 712-749
{tuln,tuongtringuyen,dosamhwang}@gmail.com)

Chang Ha Lee[2], Jason J. Jung[3]
(Department of Computer Engineering, Chung-Ang University
Seoul, Korea 156-756
{chlee,j3ung}@cau.ac.kr)

Abstract: As communities of researchers continue to become quite large and to grow incessantly, collaboration among researchers can be conducive to greater research productivity. Nevertheless, it is difficult for a researcher to find suitable collaborators from all researchers around the world. In this paper, we have used bibliographic DBLP data to extract information of a researcher and to discover the relationship between the co-authors and between authors and conferences. We evaluated some of the similarity measures and developed an innovative random walk model to find potential co-authors for a given researcher. These measures were then used to design a best model to recommend co-authors. We have also applied an HITS algorithm and proposed a ranking algorithm to rank researchers and conferences with the intent of recommending authors or conferences.

Key Words: DBLP Database; Scientists Searching; HITS algorithm; Random Walk Model

Category: H.1.1, H.3.5, I.2.11

1 Introduction

Scientific research communities continue to grow. With a large number of academic events (e.g., conferences), it is difficult to support researchers in sharing their interests and ideas. DBLP is an on-line resource that provides bibliographic information on major computer science conference proceedings and journals. There is a social network implicit in the DBLP database that includes information on authors, their papers and conferences in which they have published. DBLP is a good starting point to find authors and publications.

The publishing industry depends upon the reputation and history of a given author as criteria to decide whether to publish a manuscript. As a result it can

[1] This paper is significantly revised from an earlier version presented at The 7th Asian Conference on Intelligent Information and Database Systems (ACIIDS'2015) in March 2015.

[2] These authors contributed equally to this work as the first author.

[3] Corresponding author

be difficult for junior faculty members to break into the publishing world early in their careers. One way to begin to establish a reputation is to become affiliated with established faculty members and to collaborate on work that is likely to be published. The best example of this is to have dissertation advisor with an established reputation, but once the individual finds an academic position, the new faculty member needs to find other collaborators in order to begin to establish their own reputation as key contributors to the published dialogue.

Co-author prediction has been modeled as a similarity measuring problem, a recommendation or a classification problem [Han et al., 2013]. When viewed as a similarity measuring problem, the similarities between any two authors can be calculated, and then the pairs of authors are ranked with those in the top positions chosen as the predicted links. The Collaborative Filtering (CF) method has been extended for people-to-people recommendation. However, CF suffers from the cold-start problem when data is sparse. In addition to local and global network topological features, other features can also help improve the prediction performance. For example, author-keyword matching, publication classification code matching, and meta-paths in heterogeneous information networks have all been found to be useful. For example, coauthor prediction has been modeled as a binary classification problem in order to combine multiple features.

Multi-mode networks are needed to represent additional information, and in contrast to common one-mode networks where authors are actors and co-authorship is considered to be a relation, multi-mode networks are capable of representing relationships and affiliations, single publications, conferences, journals and so on [Sun et al., 2011]. These kinds of networks are also known to consist of affiliation or membership networks where one set of actors (authors in this case) and multiple sets of events (publications, affiliations, conferences, journals, and so on). The relationships within the bibliographical data can be listed as a hierarchy with increasing indirection:

- authors within the same publication (i.e., coauthors),
- coauthors of coauthors (i.e., friend-of-a-friend),
- authors of the same conference (journal issue) (i.e., DEXA 2006),
- authors of the same conference stream (journal) (i.e., VLDB),
- authors within similar conferences (journal),
- authors with similar publications.

It is more crucial than ever to identify relevant publications, similar authors, and conferences of interest. In academia, scientific research achievements are inconceivable without collaboration and cooperation among researchers. Research collaboration can bring different perspectives and can generate greater productivity. However, it is often difficult and time-consuming for researchers to find the appropriate collaborators when searching through large volumes of scholarly data.

In contrast to common web information systems such as Yahoo, Lycos, or Google (Scholar), bibliographical databases like DBLP or io-port.net offer much more information that cannot be directly retrieved through simple queries. The latter services provide a wealth of information, including author relationships, conferences, and the evolution of the scientific community. Although basic information, i.e., author/co-author relationships, is given directly in the bibliographical section, entity relations beyond those of the documents are not detected with conventional systems and are not accessible by the user [Klink et al., 2006].

Identifying and maintaining the appropriate collaboration relations is critical for researchers because collaboration can bring together varied expertise to the same research problems and can generate results that are more productive [Deng et al., 2008]. Many researches have presented methods for co-author prediction [Sun et al., 2011, Deng et al., 2008, Nguyen et al., 2014]. Link prediction techniques have been developed for social networks of the research community in order to predict future collaboration and to provide researchers information on possible coauthors [Han et al., 2013]. In terms of a co-authorship network among scientists, there are a number of reasons exogenous to the network for which two scientists who have never written a paper together will do so in the next few years. For example, they may happen to be located geographically close once one of either of them moves to a new academic institution. Such collaboration can be difficult to predict. However, there is also a sense that a great number of instances of new collaboration can be hinted at by the topology of the network, that is, two scientists who are close to each other in a network will have colleagues in common and will travel in similar social circles, which suggests that they themselves are more likely to collaborate in the near future [Klink et al., 2006].

In this paper, we focus on finding potential co-authors for an existing author based on the following information: papers that both authors have written together in the past; conferences to which both authors have submitted papers; keywords that both authors have used in the titles of their papers; the position of each of the authors' names in a paper; the length of time during which both authors have been working together; and the frequency with which both have cooperated.

Such information can be retrieved from the DBLP data, and the problem is then solved with the use of probability theory, similarity measures, the random walk model and a ranking algorithm. The next section introduces the basic knowledge related to these theories.

2 DBLP dataset

2.1 DBLP Bibliographical Data

As of June 2009, the DBLP Computer Science Bibliography from the University of Trier contained more than 1.2 million bibliographic records [Ley, 2002]. For researchers in computer science, the DBLP web site is a useful tool to track the work of colleagues and to retrieve bibliographic details when composing lists of references for new papers. Another use of DBLP that is sometimes controversial is to rank and profile individuals, institutions, journals, or conferences.

The DBLP data may be downloaded, and the bibliographic records are contained in a large XML file. Many researchers simply need to use non-toy files to test and evaluate their algorithms [Ley, 2002] since they are interested in XML, but not in the semantics of the data. Others inspect the DBLP data more closely because it is easy to generate several graphs, like a bipartite person-publication graph, person-journal or person-conference graphs, or a coauthor graph as an example of a social network. The methods used to analyze and visualize these medium sized graphs have been discussed in a variety of papers. Bibliometric studies are a third group of publications that make use of the full semantics of the data. The main disadvantages of DBLP for this purpose are the lack of citation information and the variation in coverage for the different subfields of computer science. The main advantages of such are their free availability and the inclusion of many conference proceedings that play an essential role in many branches of CS and have been poorly covered by other bibliographic databases. A fourth group of papers deals with person name disambiguation, which is a special aspect of data quality.

In general, there are two types of records: publication records and person records. Publication records were inspired by the BibTeX syntax and are given by one of the following elements: article (an article from a journal or magazine), inproceedings (a paper in a conference or workshop proceedings), proceedings (the proceedings volume of a conference or workshop), book (an authored monograph or an edited collection of articles), incollection (a part or chapter in a monograph), phdthesis (a PhD thesis), mastersthesis (a Master's thesis), www (a web page).

We use the DBLP [Ley, 2002, Ley, 2009] data as an example of a bibliographic network with the DBLP bibliographic network schema [Sun et al., 2011]. The network contains 4 types of objects, namely Author, Conference, Paper, and Keyword (extracted from paper title). Links exist between authors and papers and are indicated by the "writes" and "written by" relations, between papers and keywords as indicated by "contains" and "contained by", and between conferences and papers indicated with "publishes" and "published by" [Sun et al., 2011].

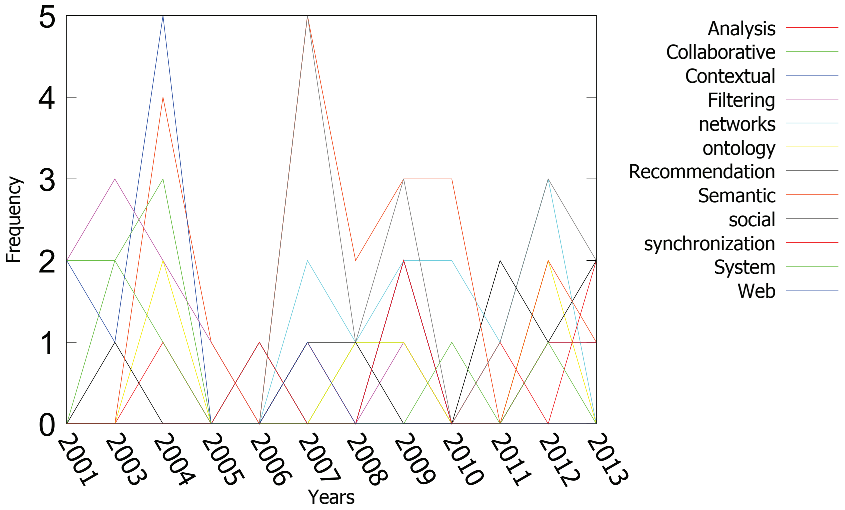

Figure 1: Some keywords extracted from paper titles of author "Jason J. Jung"

The DBLP dataset has been downloaded from the website `http://www.informatik.uni-trier.de/~ley/db/`, and the data is stored in an XML file that contains information about the conferences, journals, authors, and papers. DBLP indexes more than 18,000 journal volumes, about 20,000 conferences or workshops, more than 15000 monographs, and over 2.3 million publications, all published by more than 1.2 million authors. We downloaded the file in September 2013 and used only the publications for conferences. Our experiment did not consider any publication after that date or any journal publications.

We extracted the keywords from the paper titles in the DBLP data. In this experiment, we focused only on conference proceedings. Then we counted the frequency of every keyword through each year. We manually selected some keywords that could be considered to be topic-related words, e.g., "data", "database", "Relational", and so on. For example, some keywords extracted for the titles of the papers by author "Jason J. Jung" are shown in Fig. 1.

The extracted keywords show the fields that the author or conference has an interest in, and these let us know the research trend of the author or the conference. Based on this, we can choose an appropriate conference to which to submit the paper or can choose an appropriate author with whom to cooperate. We can also predict or recommend a future co-author who shares the same keywords as the existing author.

3 Discovering Co-author Relationship in Bibliographic Data

3.1 Using Similarity Measures

This section presents some known similarity measures that can be used to calculate the similarity between the two authors.

The Jaccard index (or Jaccard similarity coefficient) evaluates the similarity of the two sets according to the ratio of the size of the intersection of the two sets to the size of their union [Rajaraman and Ullman, 2011]:

$$sim_{p\_jac}(A,B) = \frac{|P_A \bigcap P_B|}{|P_A \bigcup P_B|} \tag{1}$$

$$sim_{c\_jac}(A,B) = \frac{|C_A \bigcap C_B|}{|C_A \bigcup C_B|} \tag{2}$$

$$sim_{k\_jac}(A,B) = \frac{|K_A \bigcap K_B|}{|K_A \bigcup K_B|} \tag{3}$$

The Soergel similarity [Cha, 2007] evaluates the similarity of the two sets according to the ratio of the size of the intersection of two sets to the maximum number of papers/conferences/keywords of two authors:

$$sim_{p\_soer}(A,B) = \frac{|P_A \bigcap P_B|}{max\left(|P_A|, |P_B|\right)} \tag{4}$$

$$sim_{c\_soer}(A,B) = \frac{|C_A \bigcap C_B|}{max\left(|C_A|, |C_B|\right)} \tag{5}$$

$$sim_{k\_soer}(A,B) = \frac{|K_A \bigcap K_B|}{max\left(|K_A|, |K_B|\right)} \tag{6}$$

The Lorentzian similarity [Deza and Deza, 2006] evaluates the similarity of two authors by the logarithm of the number of intersection of papers/conferences/keywords of two authors:

$$sim_{p\_lor}(A,B) = \ln\left(1 + \left|P_A \bigcap P_B\right|\right) \tag{7}$$

$$sim_{c\_lor}(A,B) = \ln\left(1 + \left|C_A \bigcap C_B\right|\right) \tag{8}$$

$$sim_{k\_lor}(A,B) = \ln\left(1 + \left|K_A \bigcap K_B\right|\right) \tag{9}$$

The Hamming distance between two sets is defined by the number of components in which they differ [Rajaraman and Ullman, 2011]:

$$\begin{aligned} sim_H(A,B) &= \frac{1}{1+dist_H(A,B)} \\ dist_H(A,B) &= diff_C(A,B) + diff_K(A,B) + diff_P(A,B) \\ diff_C(A,B) &= |C_A| + |C_B| - 2\left|C_A \bigcap C_B\right| \\ diff_K(A,B) &= |K_A| + |K_B| - 2\left|K_A \bigcap K_B\right| \\ diff_P(A,B) &= |P_A| + |P_B| - 2\left|P_A \bigcap P_B\right| \end{aligned} \tag{10}$$

3.2 Random Walk Model with Academic Metrics

This section introduces the Academic RWR [Li et al., 2014]. This method is based on a random walk with a restart algorithm on the author-author graph. A random walk is the process through which randomly moving objects wander away from where they started. The edge links in the graph are computed from three metrics to improve the accuracy, including co-author order, latest collaboration time point and frequency of collaboration [Li et al., 2014].

– Co-author order: is the order which the names of authors appeared in a paper. Consider two nodes p_i, p_j in a co-author list. Measure of co-author order DCL (distance in coauthor list) is calculated by:

$$DCL(p_i,p_j) = \begin{cases} \frac{1}{i} + \frac{1}{j} & \text{if } j \leq 3 \\ \frac{1}{i} + \frac{2}{j} & \text{if } j > 3, i \leq 3 \\ \frac{2}{i} + \frac{2}{j} & \text{if } i > 3 \end{cases} \tag{11}$$

– Latest collaboration time point: an author may have trend to collaborate with recent co-authors than with authors he co-authored long time ago. Measure is calculated by using $LIM_t(p_i,p_j)$ (Link Importance):

$$\begin{aligned} LIM_t(p_i,p_j) &= DCL(p_i,p_j) * k(t) \\ k(t) &= \frac{t_i - t_0}{t_c - t_0} \end{aligned} \tag{12}$$

where k(t) is a monotonically increasing function over time, t_i is the formation time, t_c is the current time, t_0 is the first link formation time.

– Times of collaboration: two authors who have collaborated many times in past may have high chance to work together again. The impact of different times of coauthoring is measured by:

$$LIM_{[t_1,t_2]}(p_i,p_j) = \sum_{t=t_1}^{t_2} LIM_t(p_i,p_j) = \sum_{t=t_1}^{t_2} DCL(p_i,p_j) * k(t) \tag{13}$$

3.3 The proposed method

Usually, two authors may have a list of common keywords, but we don't know whether those keywords are the main keywords in the conferences that the two authors have joined in. A keyword is the main keyword of a conference when that keyword is used in the titles of papers for that conference, i.e., they have a high frequency of appearance. If two authors have one common keyword and that keyword is the main keyword for the corresponding conference, they might have the share the same research direction. We therefore need to evaluate the score of the popularity or the importance of those keywords for each conference. The score of one keyword in a conference can be calculated as

$$S(k,C) = \frac{|\{k|k \in p, p \in P_C\}|}{\sum |\{k|\forall k \in p, \forall p \in P_C\}|} \tag{14}$$

where P_C is a set of the papers published in conference C.

For example, a conference C has 2 papers P_1, P_2. Paper P_1 contains keywords $\{k_1,k_2\}$, paper P_2 contains keywords $\{k_2,k_3\}$. Score of k_2 in conference C is calculated by $S_{k_2} = \frac{|\{k_2\}|}{|\{k_1,k_2,k_3\}|} = \frac{2}{4} = 0.5$. We have similarity of 2 authors A and B:

$$sim_{prop}(A,B) =$$

$$\frac{Q+1}{N} \times \frac{\sum_{\forall k_i \in K_{AB}, \forall C_j \in C_A, \forall C_l \in C_B} S(k_i,C_j) + S(k_i,C_l)}{\sum_{\forall k_i \in K_A, C_j \in C_A} S(k_i,C_j) + \sum_{\forall k_i \in K_B, C_j \in C_B} S(k_i,C_j)} \tag{15}$$

4 Ranking Algorithm

4.1 HITS Algorithm

The HITS algorithm [Kleinberg, 1999] was developed by Jon Kleinberg. The algorithm makes use of the link structure of the web in order to discover and rank pages that are relevant for a particular topic. We apply the HITS algorithm by using a set of nodes representing keywords or authors or conferences to produce an undirected graph.

We denote $A = \{a_i\}$ is a set of authors, $C = \{C_j\}$ is a set of conferences, $K = \{k_k\}$ is a set of keywords, A^{k_k} is a set of authors who used keyword k_k,

C^{k_k} is a set of conferences which contain keyword k_k, K^{a_i} is a set of keywords of author a_i, and K^{c_j} is a set of keywords of conference c_j.

We use the relationship between keywords and authors or keywords and conferences in a similar manner as hubs and authorities in [Kleinberg, 1999, Nguyen and Jung, 2015]. We use an undirected graph $G = \langle V, E \rangle$, where V is the set of nodes representing keywords or authors or conferences, and E is the set of edges [Ding et al., 2003, Benzi et al., 2012].

At the beginning, all nodes have a weight equal to 1. For each iteration, they are recomputed as follows.

– For ranking authors:

$$a_i = \sum_{j=1}^{n} \frac{1}{|A^{k_j}|} k_j; \qquad k_j = \sum_{i=1}^{m} \frac{1}{|K^{a_i}|} a_i \tag{16}$$

$$A = MK; \qquad K = M^T A \tag{17}$$

where M is an adjacency matrix $(m \times n)$ and M^T is the transpose matrix of M. The value of each elements of M, called m_{ij} is defined that:

$$m_{ij} = \frac{|P|}{freq(k_j)} \tag{18}$$

where $|P|$ is the number of papers which their titles contain keyword k_j, $freq(k_j)$ is the number of times author a_i uses this keyword k_j.

– For ranking conferences:

$$c_i = \sum_{j=1}^{n} \frac{1}{|C^{k_j}|} k_j; \qquad k_j = \sum_{i=1}^{o} \frac{1}{|K^{c_i}|} c_i \tag{19}$$

$$C = MK; \qquad K = M^T C \tag{20}$$

where M is an adjacency matrix $(o \times n)$ and M^T is the transpose matrix of M. The value of each elements of M, called m_{ij} is defined that:

$$m_{ij} = \frac{|P|}{freq(k_j)} \tag{21}$$

where $|P|$ is the number of papers which their titles contain keyword k_j, $freq(k_j)$ is the number of times keyword k_j appear in conference c_i papers titles.

This process is repeated after some iteration. For each iteration step, the values of the nodes are recomputed and normalized. When the task is complete, A and C contain the rank score of the authors and conferences. We then recommend the top ranked score of the authors or conferences. For example, we have a set of conferences $C = \{c_1, c_2, c_3\}$ and a set of keywords $K = \{k_1, k_2, k_3, k_4\}$ as described in Tab. 1.

	k_1	k_2	k_3	k_4
c_1	3	4	5	0
c_2	1	4	6	2
c_3	4	0	7	5

Table 1: Example of ranking conferences by HITS algorithm

We have the adjacency matrix M is:

$$M = \begin{bmatrix} 3\,4\,5\,0 \\ 1\,4\,6\,2 \\ 4\,0\,7\,5 \end{bmatrix}$$

We have two vectors $Conf = \{con_1,\ con_2,\ con_3\}$ and $Key = \{key_1,\ key_2,\ key_3,\ key_4\}$ contain the rank score of conferences and keywords. First, initialize $Conf = \{1, 1, 1\}$ and $Key = \{1, 1, 1, 1\}$. At the first iteration, the value of each node is recomputed as follow:$con_1 = 1.67$, $con_2 = 1.67$, $con_3 = 1.67$, $key_1 = 0.92$, $key_2 = 0.92$, $key_3 = 0.92$, $key_4 = 0.92$.

And then $Conf$ and Key are recomputed as follow:

$$Conf = M.Key = \begin{bmatrix} 3\,4\,5\,0 \\ 1\,4\,6\,2 \\ 4\,0\,7\,5 \end{bmatrix} . \begin{bmatrix} 0.92 \\ 0.92 \\ 0.92 \\ 0.92 \end{bmatrix} = (11, 11.92, 14.67)$$

$$Key = M^T.Conf = \begin{bmatrix} 3\,1\,4 \\ 4\,4\,0 \\ 5\,6\,7 \\ 0\,2\,5 \end{bmatrix} . \begin{bmatrix} 1.67 \\ 1.67 \\ 1.67 \end{bmatrix} = \begin{bmatrix} 13.33 \\ 13.33 \\ 30 \\ 11.67 \end{bmatrix}$$

The process enter the second iteration with $Conf = \{11, 11.92, 14.67\}$ and $Key = \{13.33, 13.33, 30, 11.67\}$.

4.2 AuCon-Ranking Algorithm

We proposed a ranking algorithm to rank authors or conferences, which is referred to as the AuCon-Ranking algorithm. We also denote $A = \{a_i\}$ is a set of authors, $C = \{c_j\}$ is a set of conferences, $K = \{k_k\}$ is a set of keywords, K_{a_i} is the set of keywords of author a_i, and K_{c_j} is the set of keywords of conference c_j. The rank score of an author is calculated with the following formula.

$$RS_{a_i} = \frac{\sum_{k=1}^{n} x_{ik} - \frac{1}{n}\sqrt{\sum_{k=1}^{n}(x_{ik} - \bar{x_k})^2}}{ln2^{y+1}} \tag{22}$$

where RS_{a_i} is the rank score of author a_i, n is the number of keywords in K, x_{ik} is the frequency of keyword k_k, $\bar{x_k}$ is the mean of keyword k_k, y is the number of keywords of K are missing in K_{a_i}. x_{ik} is calculated as follow:

$$x_{ik} = \frac{|P|}{freq(k_k)} \tag{23}$$

where $|P|$ is the number of papers which their titles contain keyword k_k, $freq(k_k)$ is the number of times author a_i uses this keyword k_k. $\bar{x_k}$ is calculated by formula as follow:

$$\bar{x_k} = \frac{\sum_{i=1}^{m} x_{ik}}{m} \tag{24}$$

where m is the number of authors in A. y is calculated by formula as follow:

$$y = |K| - |K_{a_i}| \tag{25}$$

For example, we have a set of authors $A = \{a_1, a_2, a_3\}$ and a set of keywords $K = \{k_1, k_2, k_3, k_4\}$ as described in Tab. 2.

	k_1	k_2	k_3	k_4
a_1	3	4	5	0
a_2	1	4	6	2
a_3	4	0	7	5

Table 2: Example of ranking authors by AuCon-Ranking algorithm

The rank score of author a_1 is calculated as follow: $\bar{x_1} = 2.67$, $\bar{x_2} = 2.67$, $\bar{x_3} = 6$, $\bar{x_4} = 2.33$, $y = 1$. Following Equ. 22, we computed $RS_{a_1} = 9.17$

Similarly, we have the formula to calculate the rank score of a conference as follow:

$$RS_{c_j} = \frac{\sum_{k=1}^{n} x_{jk} - \frac{1}{n}\sqrt{\sum_{k=1}^{n}(x_{jk} - \bar{x_k})^2}}{ln2^{y+1}} \tag{26}$$

where RS_{c_j} is the rank score of conference c_j, n is the number of keywords in K, x_{ik} is the frequency of keyword k_k, $\bar{x_k}$ is the mean of keyword k_k, y is the number of keywords of K are missing in K_{c_j}. x_{ik} is calculated as follow:

$$x_{ik} = \frac{|P|}{freq(k_k)} \tag{27}$$

where $|P|$ is the number of papers which their titles contain keyword k_k, $freq(k_k)$ is the number of times keyword k_k appear in conference c_j papers titles. $\bar{x_k}$ is calculated by formula as follow:

$$\bar{x_k} = \frac{\sum_{i=1}^{\lambda} x_{ik}}{\lambda} \tag{28}$$

where λ is the number of conferences in C, y is calculated by formula as follow:

$$y = |K| - \left|K_{c_j}\right| \tag{29}$$

5 Experimental results

5.1 Co-authors relationship

We have chosen author “Philip S. Yu”, a well-known researcher in data mining who has published more than 400 conference papers (in our collected data), as a source author. As of 2012, he had 245 coauthors, and we then continued on to identify the co-authors of those 245 authors. There are a total of 4113 authors in the graph. The data extracted from DBLP in for data mining involves 39 conferences and 38K authors. We selected data for the papers, conferences, title keywords before the year 2012. Each author was used as target author in that set to compute the similarity between source author and target author with different measures. We then compare the results with the list of the true co-authors after year 2012 for author “Philip S. Yu”.

The Academic RWR [Li et al., 2014] was implemented by building a 4113x4113 matrix called S. S is the set of probabilities for each node P_i skipping to the next node P_j. The value of each of the elements is calculated as

$$S_{i,j} = \frac{W_{i,j}}{\sum_{P_k \in N(P_i)} W_{i,k}} \tag{30}$$

where $W_{i,j}$ and $W_{i,k}$ are calculated by equation 13, $N(P_i)$ is the set of neighbors of P_i. Initialize the rank score vector MR and the restart probability vector q as (0,. . . ,1,. . . ,0), in which target node P_i is set as 1 while others are set as 0. Initialize MR vector, the rank score of a node is calculated by:

$$MR(p_i) = \frac{1-\alpha}{N} + \alpha \sum_{p_j \in M(p_i)} \frac{MR(p_j)}{L(p_j)} \tag{31}$$

$M(p_i)$ is the set of nodes incident to node p_i, $L(p_j)$ is the number of all the neighbors of node p_j, α denotes the probability of the walker continuing walking to the next node. Iterate with some step, the iterative process is defined as:

$$MR^{(t+1)} = \alpha SMR^{(t)} + (1-\alpha)q$$

where $MR^{(t)}$ is the rank score vector at step t. The rank score vector MR contains the score for each node. Finally, we get nodes in the TOP N of the list MR to recommend to target node. To evaluate the performance of these measures, we use three popular metrics [Fouss and Saerens, 2008, Shani and Gunawardana, 2011] precision, recall and F-measure. In our case, we can divide all nodes into three groups according to the following cases: collaborating with target nodes and recommended (A); collaborating with target nodes but not recommended (B); not collaborating with target nodes but recommended(C).

The metric precision (P) and recall (R) is defined as:

$$P = \frac{A}{A+C}; R = \frac{A}{A+B}; F = \frac{2PR}{P+R} \tag{32}$$

In this paper, we use the Equ. 32 to calculate the precision, recall and F-measure of each measures. The results are presented in Tab. 3, Tab. 4, Tab. 5. Besides, Fig. 2, Fig. 3 showed comparison between methods.

Notation	Meaning	Precision	Recall	F_measure
simc_jac	Jaccard Similarity (Equ. 2)	0.09	0.81	0.16
simc_soer	Soergel Similarity (Equ. 5)	0.11	0.81	0.19
simc_lor	Lorentzian Similarity (Equ. 8)	0.13	0.81	0.22

Table 3: Evaluation results by using conference information of authors

Notation	Meaning	Precision	Recall	F_measure
simk_jac	Jaccard Similarity (Equ. 3)	0.15	0.81	0.25
simk_soer	Soergel Similarity (Equ. 6)	0.13	0.81	0.22
simk_lor	Lorentzian Similarity (Equ. 9)	0.13	0.81	0.22

Table 4: Evaluation results by using keyword information of authors

Notation	Meaning	Precision	Recall	F_measure
sim_ham	Hamming distance (Equ. 10)	0.26	0.81	0.39
acrec	Academic RWR method [Li et al., 2014]	0.17	0.79	0.27

Table 5: Evaluation results from Hamming distance and Academic RWR method

We can see that the similarity measures that are computed on the number of common papers between two authors have better results than for the other measures. The best results are measures in Equ. 1, Equ. 4, Equ. 7 and Equ. 10. The second is the ACRec model [Li et al., 2014]. The result of the

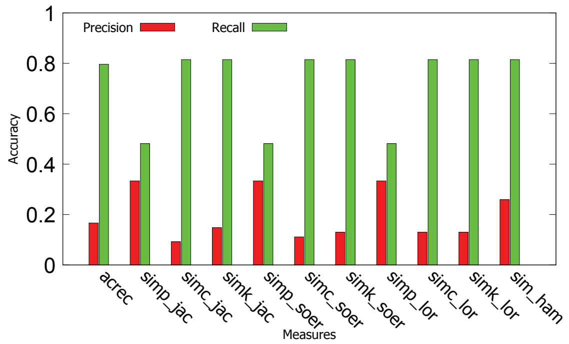

Figure 2: Precision and recall of measures

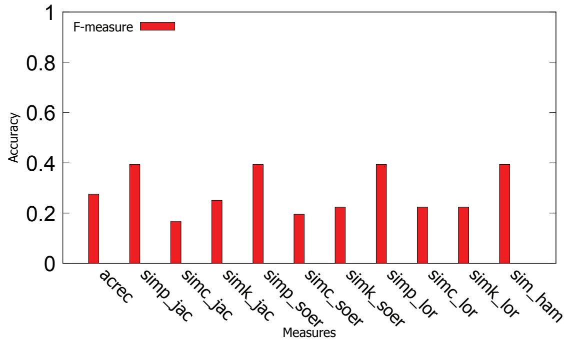

Figure 3: F-Measure

ACRec method [Li et al., 2014] is only better than the results of the similarity measures for which information is calculated with respect to the conferences or keywords. The Hamming distance computed with Equ. 10, combines three types of information: papers, conferences, and keywords. This method produces better results than the ACRec method, and in addition, the computation using similarity measures is faster than ACRec. Therefore, the use of similarity measures with information of the papers of two authors will return better results with better overall performance, especially when big data is involved.

The advantages of similarity measures are more simple in implementing and running faster. They have some disadvantages. Similarity measures calculated by Equ. 1, Equ. 4, Equ. 7 only return authors who have collaborated before. They need to combine with similarity calculated by Equ. 2, Equ. 3, Equ. 5, Equ. 6, Equ. 8 and Equ. 9 to recommend more potential co-authors who haven't collaborated before. The ACREC have advantages in finding new collaborators, but it run slowly and need more memory.

5.2 Authors and Conferences Ranking

For ranking conferences, we choose author *Jason J. Jung* and 7 keywords for experiment: *Semantic*, *social*, *Recommendation*, *Analysis*, *Data*, *Databases*, *Recommender*. The results are shown in Tab. 6 and Tab. 7. With the AuCon-Rank algorithm, we take care of number of times author *Jason J. Jung* attended in a conference before.

Conference	k_1	k_2	k_3	k_4	k_5	k_6	k_7	Rank Score
IGARSS	26	5	0	717	1999	10	0	4.33
INTERSPEECH	123	15	2	774	433	51	0	2.23
HICSS	50	329	12	453	352	33	11	2.05
ICIP	91	5	4	717	378	36	0	1.96
ICC	2	24	2	765	421	2	0	1.92
ICDE	48	24	5	56	683	304	3	1.89
GLOBECOM	3	53	4	735	393	1	1	1.88
ISCAS	9	3	0	805	345	2	0	1.83
WWW	243	279	55	106	306	24	20	1.77
CIKM	130	151	49	112	398	128	15	1.68
SEMWEB	607	39	8	26	303	18	7	1.67
KDD	28	121	25	126	597	67	23	1.64
VLDB	52	2	0	43	635	244	1	1.63
SMC	57	57	12	477	336	20	2	1.54
SIGMOD	28	18	5	58	594	217	2	1.53
ICDM	47	67	19	128	555	41	11	1.42
SAC	94	28	19	197	406	81	15	1.39
ICEIS	126	43	13	202	377	64	13	1.38
AMCIS	32	273	12	281	204	3	8	1.35
CHI	22	393	7	164	172	5	15	1.32

Table 6: Top 20 conferences ranked by HITS algorithm (k_1:Semantic, k_2: Social, k_3: Recommendation, k_4:Analysis, k_5: Data, k_6:Databases, k_7:Recommender)

There are 24 conferences in the result of ranking list that author *Jason J. Jung* attend before. We select top 24 conferences that author *Jason J. Jung* submitted most papers. Then we compare 2 lists and found out the number of matching conferences between 2 lists are 19 conferences.

For ranking authors, we choose conference SIGMOD and 6 keywords for experiment: *data*, *database*, *Databases*, *Distributed*, *Mining*, *Analysis*. The results are shown in Tab. 9. With the AuCon-Rank algorithm, we also take care of number of times that each author attended in conference SIGMOD before.

There are 74 authors in the result of the ranking list who had previously attended the SIGMOD conference. We selected the top 74 authors who had submitted the most papers in SIGMOD. We then compare both lists, and the number of authors matching on both lists is 22. Tab. 6 shows the result of the top 20 ranking conferences by using the HITS algorithm. As we can see, the IGARSS conference is ranked as number one, even though it does not have the

Conference	k_1	k_2	k_3	k_4	k_5	k_6	k_7	Rank Score
HICSS	50	329	12	453	352	33	11	1427.29
GLOBECOM	3	53	4	735	393	1	1	1335.89
SMC	57	57	12	477	336	20	2	1293.74
ICDE	48	24	5	56	683	304	3	1263.88
WWW	243	279	55	106	306	24	20	1198.02
ICCS	53	13	5	249	313	18	2	1154.95
CIKM	130	151	49	112	398	128	15	1143.07
SEMWEB	607	39	8	26	303	18	7	1128.27
KDD	28	121	25	126	597	67	23	1121.98
AMCIS	32	273	12	281	204	3	8	1105.52
KES	97	36	28	203	260	30	6	1046.78
SIGMOD	28	18	5	58	594	217	2	1037.97
IGARSS	26	5	0	717	1999	10	0	1010.59
ICDM	47	67	19	128	555	41	11	985.02
ICEIS	126	43	13	202	377	64	13	972.88
SAC	94	28	19	197	406	81	15	972.07
HCI	33	234	19	320	169	7	7	912.21
CHI	22	393	7	164	172	5	15	887.24
FSKD	58	21	17	312	328	24	2	879.52
ICMCS	121	47	18	285	208	45	3	849.48

Table 7: Top 20 conferences ranked by AuCon-Rank algorithm (k_1:Semantic, k_2: Social, k_3:Recommendation, k_4:Analysis, k_5:Data, k_6:Databases, k_7:Recommender)

Author Name	k_1	k_2	k_3	k_4	k_5	k_6	Rank Score
Jiawei Han	76	16	40	3	141	33	14.62
Philip S. Yu	107	21	13	39	78	22	12.05
Shusaku Tsumoto	52	2	26	0	58	14	6.90
Christos Faloutsos	41	4	13	1	68	14	6.64
Elisa Bertino	63	29	34	20	3	6	6.27
Hans-Peter Kriegel	51	18	43	7	17	6	6.05
Masaru Kitsuregawa	36	23	6	4	40	12	5.47
Jian Pei	50	1	7	1	57	2	5.44
Bhavani M. Thuraisingham	65	13	5	10	29	8	5.38
Reda Alhajj	34	11	16	2	42	10	5.27
Rakesh Agrawal	33	23	19	2	34	0	5.12
David Taniar	34	29	11	14	23	3	5.02
Wil M. P. van der Aalst	11	0	0	5	65	19	5.01
Divyakant Agrawal	64	18	25	18	0	3	5.00
Sushil Jajodia	47	18	31	15	5	6	4.99
Gagan Agrawal	77	0	2	14	26	7	4.98
Sang Hyuk Son	21	43	21	28	0	3	4.96
Srinivasan Parthasarathy	38	0	8	12	40	8	4.72
Alok N. Choudhary	61	4	1	27	14	12	4.62
Fosca Giannotti	34	3	13	0	39	13	4.61

Table 8: Top authors ranked by HITS algorithm (k_1:Data,k_2: Database, k_3:Databases, k_4:Distributed, k_5:Mining, k_6:Analysis)

Author Name	k_1	k_2	k_3	k_4	k_5	k_6	Rank Score
Jiawei Han	76	16	40	3	141	33	1518.39
Philip S. Yu	107	21	13	39	78	22	1224.34
Hans-Peter Kriegel	51	18	43	7	17	6	570.23
Christos Faloutsos	41	4	13	1	68	14	556.21
Hector Garcia-Molina	47	17	9	22	1	2	415.16
Elisa Bertino	63	29	34	20	3	6	399.88
Umeshwar Dayal	41	24	3	10	8	10	390.81
Jian Pei	50	1	7	1	57	2	344.94
H. V. Jagadish	22	18	9	5	5	2	324.03
Elke A. Rundensteiner	50	6	9	8	5	5	316.22
Gerhard Weikum	23	25	2	11	1	7	307.22
Sushil Jajodia	47	18	31	15	5	6	292.51
Michael Stonebraker	55	16	8	13	0	5	292.49
Divyakant Agrawal	64	18	25	18	0	3	268.15
Masaru Kitsuregawa	36	23	6	4	40	12	265.13
Rakesh Agrawal	33	23	19	2	34	0	257.73
Wolfgang Lehner	61	21	7	4	7	4	245.05
Jianzhong Li	62	8	7	7	6	2	215.31
David J. DeWitt	12	33	6	4	0	5	193.75
Michael J. Carey	35	22	5	8	1	0	186.15

Table 9: Top authors ranked by AuCon-Rank algorithm (k_1:Data,k_2: Database, k_3:Databases, k_4:Distributed, k_5:Mining, k_6:Analysis)

keyword “Recommendation” and “Recommender” in its set of keywords while HICSS has all keywords in its set and is ranked as number three. When using the AuCon-Ranking algorithm to rank conferences, as we can see in the Tab. 7, IGARSS is ranked as number thirteen while HICSS is ranked as number one because HICSS has a good distribution of keywords in its set of keywords. This the main difference between using the AuCon-Ranking and HITS algorithm, and we appreciate the AuCon-Ranking.

6 Conclusion and Future Work

In this paper, we have introduced co-author relationships in bibliographic data. We present the results of an experiment using the DBLP dataset. DBLP is a good starting point to discover information related to authors, journals and conferences. We then extract information from DBLP about the authors, including the number of publications, number of conferences attended, and the keywords that the authors have used in their paper titles. We can also obtain information about the conferences, such as number of publications and authors who have published papers in them.

The research community is large and continues to grow, so we have introduced various similarity measures that can be used to find similar authors. These similarity measures are applied with a variety of information related to the authors, including the number of papers, conferences, and keywords. The results of the experiments show that similarity measures computed for the number of common papers for two authors provide better metrics for evaluation than others. This means that an author has a tendency to collaborate with previous co-authors. The other two metrics, including the number of common conferences and the number of common keywords, may be helpful to recommend new co-authors. The other method involves a random walk model, which is also efficient in finding new neighbors.

We also studied the issue of recommending authors and conferences by using ranking algorithms. We introduced the HITS algorithm and proposed a ranking algorithm, AuCon-Rank. The two algorithms ranked authors or conferences according to a set of keywords and recommend the top-ranked authors or conferences. The AuCon-Rank method is simple and the distribution in the ranked results is also better than that of the HITS algorithm.

The advantages of the similarity measures are even simpler with respect to the implementation and to a speedy execution. However, there are some some disadvantages to the use of such. Similarity measures calculated by Equ. 1, Equ. 4, Equ. 7 only return authors who have collaborated before. These need to be combined with similarity calculated by Equ. 2, Equ. 3, Equ. 5, Equ. 6, Equ. 8 and Equ. 9 in order to recommend more potential co-authors who haven't collaborated before. ACREC has the advantages in finding new collaborators, but runs slowly and requires more memory. The HITS and the ranking algorithm are the only focus for the keywords, and these need to include more information in order to improve the accuracy.

We later need to find a model to combine the best measures in this work so that a new model can be produced. The new model should include more information of the author to improve accuracy and could provide better results for the recommendation, not only for old co-authors, but also for new potential co-authors.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. NRF-2013K2A1A2055213). Also, this research was supported by the MSIP (Ministry of Science, ICT&Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (NIPA-2014-H0301-14-1044) supervised by the NIPA (National ICT Industry Promotion Agency).

References

[Benzi et al., 2012] Benzi, M., Estrada, E., and Klymko, C. (2012). Ranking hubs and authorities using matrix functions. *CoRR*, abs/1201.3120.

[Cha, 2007] Cha, S.-H. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 1(4):300–307.

[Deng et al., 2008] Deng, H., King, I., and Lyu, M. R. (2008). Formal models for expert finding on DBLP bibliography data. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy*, pages 163–172.

[Deza and Deza, 2006] Deza, E. and Deza, M. (2006). *Dictionary of Distances.* North-Holland.

[Ding et al., 2003] Ding, C. H. Q., He, X., Husbands, P., Zha, H., and Simon, H. D. (2003). Pagerank: HITS and a unified framework for link analysis. In *Proceedings of the Third SIAM International Conference on Data Mining, San Francisco, CA, USA, May 1-3, 2003*, pages 249–253.

[Fouss and Saerens, 2008] Fouss, F. and Saerens, M. (2008). Evaluating performance of recommender systems: An experimental comparison. In *2008 IEEE / WIC / ACM International Conference on Web Intelligence, WI 2008, 9-12 December 2008, Sydney, NSW, Australia, Main Conference Proceedings*, pages 735–738.

[Han et al., 2013] Han, S., He, D., Brusilovsky, P., and Yue, Z. (2013). Coauthor prediction for junior researchers. In *Social Computing, Behavioral-Cultural Modeling and Prediction - 6th International Conference, SBP 2013, Washington, DC, USA, April 2-5, 2013. Proceedings*, pages 274–283.

[Kleinberg, 1999] Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632.

[Klink et al., 2006] Klink, S., Reuther, P., Weber, A., Walter, B., and Ley, M. (2006). Analysing social networks within bibliographical data. In *Proceedings of the 17th International Conference on Database and Expert Systems Applications (DEXA 2006), Kraków, Poland, September 4-8*, pages 234–243.

[Ley, 2002] Ley, M. (2002). The dblp computer science bibliography: Evolution, research issues, perspectives. In *String Processing and Information Retrieval, 9th International Symposium, SPIRE 2002, Lisbon, Portugal, September 11-13, 2002, Proceedings*, pages 1–10.

[Ley, 2009] Ley, M. (2009). Dblp - some lessons learned. *PVLDB*, 2(2):1493–1500.

[Li et al., 2014] Li, J., Xia, F., Wang, W., Chen, Z., Asabere, N. Y., and Jiang, H. (2014). Acrec: a co-authorship based random walk model for academic collaboration recommendation. In *23rd International World Wide Web Conference, WWW '14, Seoul, Republic of Korea, April 7-11, 2014, Companion Volume*, pages 1209–1214.

[Nguyen et al., 2014] Nguyen, T. T., Hwang, D., and Jung, J. J. (2014). Social tagging analytics for processing unlabeled resources: A case study on non-geotagged photos. In *Intelligent Distributed Computing VIII - Proceedings of the 8th International Symposium on Intelligent Distributed Computing, IDC 2014, Madrid, Spain, September 3-5, 2014*, pages 357–367.

[Nguyen and Jung, 2015] Nguyen, T. T. and Jung, J. J. (2015). Exploiting geotagged resources to spatial ranking by extending hits algorithm. *Computer Science and Information Systems*, 12(1):185–201.

[Rajaraman and Ullman, 2011] Rajaraman, A. and Ullman, J. D. (2011). *Mining of Massive Datasets.* Cambridge University Press, New York, NY, USA.

[Shani and Gunawardana, 2011] Shani, G. and Gunawardana, A. (2011). *Evaluating Recommendation Systems.* Springer.

[Sun et al., 2011] Sun, Y., Barber, R., Gupta, M., Aggarwal, C. C., and Han, J. (2011). Co-author relationship prediction in heterogeneous bibliographic networks. In *International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2011, Kaohsiung, Taiwan, 25-27 July 2011*, pages 121–128.