Expert Systems with Applications 39 (2012) 9788–9796

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

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

PAV: A novel model for ranking heterogeneous objects in bibliographic information networks

Zhi-Hong Deng [a,b,*], Bo-Yan Lai [a], Zhong-Hui Wang [a], Guo-Dong Fang [a]

[a] Key Laboratory of Machine Perception (Ministry of Education), School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China
[b] The State Key Lab of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China

ARTICLE INFO

Keywords:
Bibliographic information networks
Link analysis
Ranking
Regular Markov chain

ABSTRACT

Bibliographic information networks, formed by online bibliographic databases, such as ACM Digital Library and IEEE/IET Electronic Library, contain abundant information about authors, papers, venues (journals/conferences), and have been widely studies in recent years. However, few studies examine the problem of ranking objects in these networks. In this paper, we study this problem and present a novel model, called PAV, for ranking heterogeneous objects, such as authors, papers, and venues. Based on PAV model, we transform the problem of ranking objects into the problem of estimating probability distribution. We propose an efficient algorithm to estimate probability parameters by use of the fact that the PAV model is a regular Markov chain. For evaluating PAV model, we apply it on one real dataset, which was crawled from ACM Digital Library. The experimental results show that the proposed model is effective.

1. Introduction

There are a large number of online bibliographic databases, such as ACM Digital Library,[1] IEEE/IET Electronic Library,[2] DBLP,[3] Citeseer,[4] and Google Scholar[5] in computer science and PubMed[6] in medical sciences. Each such database indicates a tremendous information network, in which authors, papers, and venues (journals/conferences) are interconnected. Such network is called bibliographic information networks (Sun, Yu, & Han, 2009a) and has been widely studied in recent years. The main components of a bibliographic information network are three types of objects: authors, venues, and papers. In term of topological structure, a bibliographic network is a graph, in which vertices represent objects and edges represent the links between objects. In bibliographic information network, Links exist between papers and authors by the relation of "write" and "written by", papers and terms by the relation of "cite" and "cited by", papers and venues by the relation of "publish" and "published by". Fig. 1 shows a simple example.

Currently, research on bibliographic information networks has mainly focused on the implementation and management of these systems (Hwang, Hristidis, & Papakonstantinou, 2006; Sun et al., 2008; Tang et al., 2008) and the use of data mining technologies for tasks such as authority-based keyword search (Balmin & Hristidis, 2004), clustering (Sun et al., 2009a, 2009b; Yin, Han, & Yu, 2006), topic modeling (Sun, Han, Gao, & Yu, 2009; Tang & Jin, 2008), social network Extraction (Tang, Zhang, & Yao, 2007) and relationship mining (Wang et al., 2010). However, there has been little research on ranking aspects of objects in bibliographic information networks.

The exploration of object ranking in a bibliographic information network may plays a key role in searching, recommending, and mining bibliographic information, such as placing important objects (papers or authors) in the top of query results, recommending valuable papers to researchers. Due to the intrinsic heterogeneity of bibliographic information networks (Sun et al., 2009a), traditional ranking methods aiming at homogeneous data, such as methods for ranking journals in informetrics and methods for ranking web pages in search engine, are unsuitable for ranking objects in bibliographic information networks.

The above discussions motivate us to study the problem of ranking objects in bibliographic information networks. In this paper, we propose a unified model, PAV, for ranking heterogeneous objects, such as papers, author, and venues. According to this model, a bibliographic information network is represented by a weighted directed graph, where a vertex stands for an object, an edge stands for the link between objects, and a weight over an edge

* Corresponding author at: Key Laboratory of Machine Perception (Ministry of Education), School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China.
E-mail address: zhdeng@cis.pku.edu.cn (Z.-H. Deng).

[1] http://portal.acm.org.
[2] http://ieeexplore.ieee.org/Xplore/.
[3] http://dblp.uni-trier.de/.
[4] http://citeseer.ist.psu.edu/.
[5] http://scholar.google.com/.
[6] http://www.ncbi.nlm.nih.gov/pubmed/.

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2012.02.175

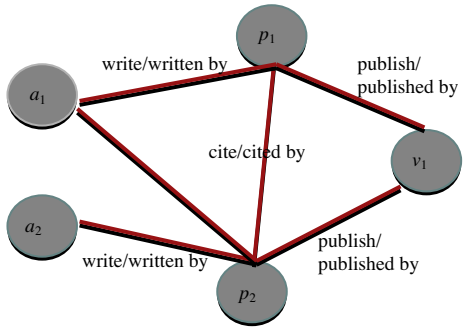

Fig. 1. An example of bibliographic information networks.

stands for the degree of contribution that one object devotes to the importance or reputation of the corresponding object sharing the same edge with the object. Based on the model, we assume that the rank (importance or reputation) of an object is the probability that the corresponding vertex is accessed by random walk in the PAV graph. For estimating the above probability, we then give an efficient resolution by utilizing the fact that the PAV model is a regular Markov chain.

As far as we know, this paper is the first study that simultaneously ranks heterogeneous objects in an unified framework. On all accounts, the contribution of this paper can be summarized as follows:

(1) We propose a novel model called PAV to capture the intrinsic relationships of different objects in bibliographic information networks, and an efficient solution is provided to rank authors, papers, and venues.
(2) Experiments, conducted on one real dataset, show the effectiveness of the proposed model in comparison with the existing rank systems available on the Web.

The rest of paper is organized as follows. Section 2 is an introduction to related work. In Section 3, we formally introduce the PAV model and several relevant concepts. In Section 4, we systematically develop a solution, which is based on the PAV model and regular Markov chain, for ranking objects. Section 5 is the experiment study and Section 6 concludes this study and points out future works.

2. Related work

Object ranking has been extensively studied for decades in informetrics (or bibliometrics), information retrieval, and digital library. A lot of approaches (Bensman & Wilder, 1998; Brin & Page, 1998; Burrell, 2007; Egghe, 2006a, 2006b; Frandsen & Rousseau, 2005; Garfield, 1972, 1998; Hirsch, 2005; Kleinberg, 1999) have been proposed to rank different kinds of objects, such as journals, authors, and web pages.

The impact factor is a dominant method for ranking journals. It evaluates a journal by comparing the number of papers, which it published in the last two years, with the number of papers, which have cited the papers published by it in the last two years. Since the impact factor was first introduce in 1972, it has been widely used by many organizes, such as Institute of Scientific Information[7] (ISI). While the original impact factor uses a two-year time window, some solutions (Frandsen & Rousseau, 2005; Garfield, 1998) were proposed to evaluate journals by using different periods of time.

In 2005, Hirsch introduced the *h*-index (Hirsch, 2005) as a new indicator for evaluating scientists. This index is defined as the highest rank on a scientist's list of publications such that the first *h* publications received at least *h* citations. Different from the impact factor, the *h*-index cannot decrease for a given date set. Therefore, it can be considered as an accumulating indicator for lifetime achievement in the case of individual scientists. As a major tool for evaluating scholars, *h*-index is also widely used in many organizes, such as Microsoft Academic Search[8] and Arnetminer.[9] Based on *h*-index, a lot of more complex indexes, such as *g*-index (Egghe, 2006a, 2006b) and *h*-rate (Burrell, 2007), have been proposed.

PageRank (Brin & Page, 1998), first introduced in 1998, is a renowned method for ranking Web pages and is a trademark of Google. PageRank regard the importance of a Web page as the likelihood that a person randomly clicking on links will arrive at it. Based on the graph created by Web pages as nodes and hyperlinks as edges, PageRank employed a random walk model to compute the probability distribution of nodes in the graph. The importance of a Web page is the probability of its corresponding node. In almost the same year, HITS (Kleinberg, 1999), another famous ranking method, was proposed. The idea (Manning, Raghavan, & Schutze, 2008) behind Hub and Authority, two core concepts of HITS, originated from the following assumptions: certain web pages, known as hubs, served as large directories that were not actually authoritative in the information that it held, but were used as compilations of a broad catalog of information that led users to other authoritative pages directly. That is, a good hub represented a page that pointed to many other pages, and a good authority represented a page that was linked by many different hubs. The importance of a Web page is determined by its authority value or hub value.

However, these above methods have a common characteristic that they only focus on one kind of objects. For example, *h*-index and *g*-index aim at ranking authors and the impact factor aim at ranking journals. This characteristic limits these methods to consider various relations among authors, papers, and venues sufficiently. The existing methods for ranking authors, such as *h*-index and *g*-index, evaluate the reputation of an author just by the papers he wrote. The impact factor evaluates the reputation of journals just by cite-relations among papers published by them. Both PageRank and HITS also evaluate the reputation of a Web page just by the Web pages that link to or link from it.

Based on the above discussion, we provide a novel model, PAV, which takes into account all relationships that exist in authors, papers, and venues. The PAV model captures five kinds of links, which are author-author, author-paper, author-venue, paper-paper, and paper-venue. Based on the PAV model, we develop an efficient solution for ranking author, paper, and venues simultaneously. In our method, the importance or reputation of an author is influenced by his co-authors, his papers, and the venues that published his papers. The importance or reputation of a paper is influenced by its authors, its venue, and the papers that cited it. The importance or reputation of a venue is influenced by the papers that it published and the authors who had papers published by the venue.

Markov chain (Grinstead & Snell, 1997), is a mathematical method to model random variables transitions from one state to another in a chainlike manner. The distinguished characteristic of Markov chain is that the next state depends only on the current state and not on the entire past. Markov chains have many applications as statistical models of real-world processes and are applied in a number of ways to many different fields, such as information sciences (Trivedi, 2002), chemistry (Peter, David, & Eugene, 2009), economics and finance (James, 1989). In this paper, we will present

[7] http://science.thomsonreuters.com.
[8] http://academic.research.microsoft.com.
[9] http://www.arnetminer.org.

a novel Markov chain model that efficiently captures various link structures in bibliographic information networks. Based on the model, we finally develop an efficient solution for ranking various objects.

3. The PAV model

In this section, we will firstly describe PAV model, which is the abbreviation for the Paper–Author–Venue graph model. Over the PAV model, we then define a transition probability matrix, abbreviated as TPM, in which an element represents the probability that one travels from one vertex (or object) to another vertex..

3.1. Preliminaries

Academic objects, like authors, papers and venues, have rich associations between each other. For example, a paper is usually written by authors working together, cites some previous papers, and finally will usually be published by a journal (or conference). Currently, most popular academic object ranking strategies make use of only part of those relationships. If all those information can be fully introduced into ranking, we can expect a more rational result. For better understanding some concepts, we first introduce the following example.

Example 1. Fig. 1 shows a simple example of bibliographic information network. In this network, there are two authors: a_1 and a_2, two papers: p_1 and p_2, and one journal v_1. The relationships between them are:

- a_1 write p_1 alone.
- a_2 and a_1 work together to write p_2. We assume that a_1 is the second author and a_2 is the first author.
- p_2 is cited by p_1.
- p_1 and p_2 are both published by v_1.

How to exploit all those information to rank effectively? Next, we will propose PAV, a novel model to solve this problem.

In this paper, a PAV model is a weighted directed graph defined as follows.

Definition 1 (*PAV model*). Let $A = \{\text{authors}\}$, $P = \{\text{papers}\}$, and $V = \{\text{Venues}\}$ are three types of objects, a weighted directed graph $G = \langle N, E, W\rangle$ is called a PAV model on objects $A \cup P \cup V$, if

(1) N, the set of vertexes in G, is equal to $A \cup P \cup V$.
(2) E, the set of edges in G, is constructed by the following operations.

- If an author a wrote a paper p, we add two directed edges $e(a,p)$ and $e(p,a)$ to link vertex a and vertex p. $e(a,p)$ represents an edge from a to p while $e(p,a)$ represents an edge from p to a.
- Let a_1 and $a_2$are two authors. If a_1 and $a_2$are the co-author of some papers, we add two directed edges $e(a_1,a_2)$ and $e(a_2,a_1)$ to link vertex a_1 and vertex a_2. $e(a_1,a_2)$ represents an edge from a_1 to a_2 while $e(a_2, a_1)$ represents an edge from a_2 to a_1.
- Let a be an author and v be a venue. If a wrote papers published by v, we add two directed edges $e(a, v)$ and $e(v,a)$ to link vertex a and vertex v. $e(a, v)$ represents an edge from a to v while $e(v,a)$ represents an edge from v to a.
- Let p be a paper and v be a venue. If p was published by v, we add two directed edges $e(p, v)$ and $e(v,p)$ to link vertex p and vertex v. $e(p, v)$ represents an edge from p to v while $e(v,p)$ represents an edge from v to p.
- Let p_1 and p_2 are two papers. If p_1 cited p_2, we add a directed edge $e(p_1,p_2)$, which represents an edge from p_1 to p_2, links vertex p_1 and vertex p_2.

(3) For each edge in E, we attach a weigh w to denote the contribution of one vertex devoting to the importance or reputation of the vertex to which the edge links the former vertex. All such weighs make up W.

As stated in Definition 1, there are five types of edges, which correspond to the links between author-author, author-paper, author-venue, paper-paper, and paper-venue. We do not consider the relationship between different journals because no explicit or meaningful links between them.

In Definition 1, the weight of an edge is not defined quantitatively. Obviously, W is most critical if we want to use a PAV model to rank objects. In this paper, we define these weights according to the type of the corresponding edges as follows.

Definition 2. Let $e(a,p)$ and $e(p,a)$ be two edges that link author a and paper p. the weight $w_{a,p}$ and $w_{p,a}$, which correspond to $e(a,p)$ and $e(p,a)$ respectively, are defined as

$$w_{a,p} = w_{p,a} = \frac{\frac{1}{s_{p,a}}}{\sum_{a' \in A(p)} \frac{1}{s_{p,a'}}} \tag{1}$$

$s_{p,a}$ is the place of a in the author list of p. for example, if a is the first author of p, $s_{p,a}$ is equal to 1. $A(p)$ means the set of authors of p.

Let's consider example 1. $A(p_2)$ is $\{a_2, a_1\}$ and $s_{p2,a1}$ is equal to 2. Formula (1) means that the front authors of a paper have closer association with the paper more than the others. Meanwhile, formula (1) represents that when these are more authors in a paper, the associations between the paper and its authors will be lower.

Definition 3. Let $e(a_1,a_2)$ and $e(a_2,a_1)$ be two edges that link author a_1 and author a_2. The weight $w_{a1,a2}$ and $w_{a2,a1}$, which correspond to $e(a_1,a_2)$ and $e(a_2,a_1)$ respectively, are defined as

$$w_{a1,a2} = w_{a2,a1} = \sum_{p:a1 \in A(p) \wedge a2 \in A(p)} \frac{1}{s_{p,a1} \times s_{p,a2}} \tag{2}$$

where the meanings of $s_{,p,a}$ and $A(p)$ is the same as in Definition 2.

When calculating $w_{a1,a2}$ and $w_{a2,a1}$, we consider all papers that a_1 and a_2 wrote together. For each paper, their places in the paper are multiplied. Hence, the inter-impact between them relies on the place they are in the author list of the paper. Obviously, the upper they are in the author list of a paper, the bigger their inter-impact is in the paper. It accords with our intuition.

Definition 4. Let $e(a, v)$ and $e(v,a)$ be two edges that link author a and venue v. The weight $w_{a,v}$ and $w_{v,a}$, which correspond to $e(a, v)$ and $e(v,a)$ respectively, are defined as

$$w_{a,v} = w_{v,a} = \sum_{p:p \in P(v) \wedge a \in A(p)} w_{a,p} \tag{3}$$

where $A(p)$ is the same as in Definition 2. $P(v)$ denotes the set of papers published by v. Considering Example 1, $P(v_1)$ is $\{p_1,p_2\}$. The Formula (3) means that the inter-impact between an author and a venue is based on the papers that both wrote by the author and published by the venue.

Definition 5. Let $e(p, v)$ and $e(v,p)$ be two edges that link paper p and venue v. The weight $w_{p,v}$ and $w_{v,p}$, which correspond to $e(p, v)$ and $e(v,p)$ respectively, are defined as

$$w_{p,v} = w_{v,p} = 1 \tag{4}$$

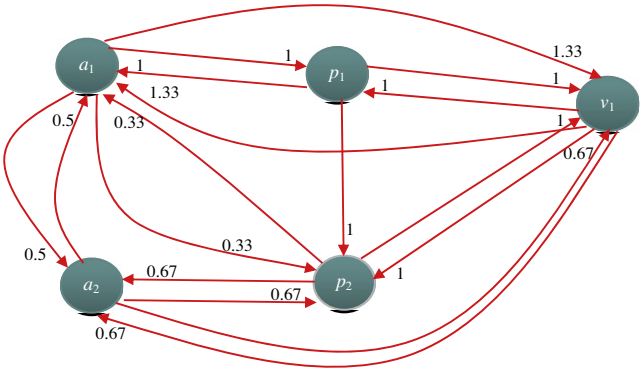

Fig. 2. The PAV model of the academic information network showed by Fig. 1.

Consider Example 1, $w_{p1,v}$, $w_{v,p1}$, $w_{p2,v}$, and $w_{v,p2}$ are all equal to 1.

Definition 6. Let $e(p_1,p_2)$ be a edge that link paper p_1 and paper p_2. The weight $w_{p1,p2}$, corresponding to $e(p_1,p_2)$, are defined as

$$w_{p1,p2} = 1 \tag{5}$$

We simply set $w_{p1,p2} = 1$ when paper p_2 was cited by paper p_1.

Based on Definition 1, we can transfer a bibliographic information network to a PAV graph model. Let's consider the bibliographic information network illustrated by Fig. 1. First, we can construct the graphic structure of the PAV model by mapping objects to vertexes and links to edges. Then, based on Formula (1)–(5), we can calculate weighs as follows.

- $w_{a1,p1} = w_{p1,a1} = \frac{1/1}{1/1} = 1$
- $w_{a1,p2} = w_{p2,a1} = \frac{\frac{1}{1}}{\frac{1}{1}+\frac{1}{2}} = \frac{2}{3} = 0.67$
- $w_{a2,p2} = w_{p2,a2} = \frac{\frac{1}{2}}{\frac{1}{1}+\frac{1}{2}} = \frac{1}{3} = 0.33$
- $w_{a1,a2} = w_{a1,a2} = \frac{1}{s_{p2,a1}\times s_{p2,a2}} = \frac{1}{1\times 2} = \frac{1}{2} = 0.5$
- $w_{a1,v1} = w_{v1,a1} = w_{a1,p1} + w_{a1,p2} = 1 + 0.67 = 1.67$
- $w_{a2,v1} = w_{v1,a2} = w_{a2,p2} = 0.33$
- $w_{p1,v1} = w_{v1,p1} = 1$
- $w_{p2,v1} = w_{v1,p2} = 1$
- $w_{p1,p2} = 1$

Final, we get the PAV model illustrated by Fig. 2.

4. Ranking objects by the PAV model

In this section, we introduce our solution that ranks objects by a PAV model. In fact, we regard the importance of reputation of an object as the probability that one access to the vertex representing the object by randomly walking in the PAV model. For evaluating the probability, we first introduce the TPM, abbreviating for transition probability matrix.

4.1. Transition probability matrix

How can we use the PAV model, in which the relationships between academic objects are well defined, to effectively get the importance or reputation of all those objects?

We consider the PAV model as a random walking model in which the viewers walk through one vertex to another by the chance that is directly proportional to the edge weight. For example, after a viewer finish reading paper p, he may want to continue reading those papers cited by p for acquainting more relevant information. For the same reason, he will probably search the authors or the journal by which paper p was published. Meanwhile, the viewer has a chance to jump out and pick up any academic object randomly at any time. In the random walking world, those vertexes, which will be visited more frequently, have the bigger value in importance or reputation.

We denote the transform probability from a vertex i to another vertex j as$pr_{i\to j}$. For a PAV model, we define$pr_{i\to j}$ as follows:

$$pr_{i\to j} = \frac{\varepsilon}{|N|} + \frac{(1-\varepsilon)}{\sum_{e(i,k)\in E} w_{i,k}} \times w_{i,j}, \quad 0 < \varepsilon < 1 \tag{6}$$

In formula (6), ε is the probability of random jumping out. $|N|$ is the number of vertexes. Therefore, $\varepsilon/|N|$ is the probability of random jumping out from vertex i to vertex j.

The second part of formula (6) measures the probability of visiting vertex j by a viewer after he/she visited vertex i. In this part, the denominator is the sum of weight of edges that link vertex i to other vertexes.

All those $pr_{i\to j}$ form a transition probability matrix, TPM. We denoted a TPM as M_{pr}. Suppose the PAV model contains n vertexes, and we have:

$$M_{pr} = \begin{Bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ p_{n1} & \cdots & \cdots & p_{nn} \end{Bmatrix}$$

For example, we can calculate the TPM of the PAV model which is shown in Fig. 2 if ε is set to be 0.2. The result is shown in Table 1.

4.2. Solution for ranking objects

In this paper we proposed the PAV model and a ranking algorithm based on the PAV model. After we get the transition probability matrix from the PAV model, an iterative computing process is introduced to get the ranking value of all objects in the PAV graph. When the procedure comes to the predefined condition, it stops. The pseudocode of our method is illustrated as follows:

Algorithm 1. Ranking_PAV

Input: a PAV graph, $G = \langle N,E,W\rangle$, M_{pr}, the transition probability matrix of G, and two threshoulds, ε and ξ. Note that N is the set of vertexes, E is the set of edges, and W is the set of weights.
Output: the ranking values of all vertexes.
Procedure:
Vec_C ← a vector of length $|N|$;
Vec_R ← a vector of length $|N|$;
Initialize each element C_i in *Vec_C*
  $C_i \leftarrow p_i$ with the restraints of $0 \leqslant p_i \leqslant 1$ and $\sum p_i = 1$;
Vec_R ← *Vec_C* × M_{pr};
While $\|Vec\_R - Vec\_C\| > \xi$ do
  Vec_C ← *Vec_R*;
  Vec_R ← *Vec_C* × M_{pr};
End while
Output *Vec_R*;

Let's consider the PAV graph showed by Fig. 2. By setting the vector of the initial probability distribution showed by Table 2, we can get the constant final probability distribution after running 45 times iteratively. Table 3 shows the result. Clearly, the result consists with our intuitions. We know that p_1 cited p_2 and no other citation-relation in the graph. Therefore, we may draw the conclusion that p_2 is more important or renowned than p_1.

Table 1
The TPM of the PAV model in Fig. 2.

	A_1	A_2	P_1	P_2	J_1
A_1	0.04	0.26	0.26	0.11	0.33
A_2	0.37	0.04	0.04	0.27	0.27
P_1	0.31	0.04	0.04	0.31	0.31
P_2	0.16	0.32	0.04	0.04	0.44
J_1	0.30	0.18	0.24	0.24	0.04

Table 2
The initial probability distribution.

a_1	a_2	p_1	p_2	v_1
1	0	0	0	0

Table 3
The final probability distribution.

a_1	a_2	p_1	p_2	v_1
0.2285	0.1798	0.1430	0.1876	0.2611

Clearly, the critical problem of our method is that problem of convergence. To ensure the defendable of ranking result, our method must be convergent. That is to say, no matter which original probability distribution is, our method must generate a stable result. In the following subsection, we will prove that our method is convergent.

4.3. Convergence of ranking_PAV

In the subsection, we discuss the problem of convergence. Before giving our proof, we first introduce some relevant concepts.

4.3.1. Markov chain and regular Markov chain

A Markov chain is a mathematical model that undergoes transitions from one state to another in a chainlike manner. It is a stochastic process that the next status is depended on the current status.

Definition 7. Consider a stochastic process $\{X_1,X_2,\ldots\}$ that takes on a finite number of possible values (or states) which denotes by $\{1,2,\ldots,r\}$. If $X_i = j$, the process is said to be in state j at time i. A stochastic process is called as Markov chain if

$$P(X_{n+1} = j|X_n = i, X_{n-1} = i_{n-1}, \ldots X_1 = i_1) = P(X_{n+1} = j|X_n = i) = p_{ij} \tag{8}$$

That is to say that when the process is in state i, there is a fixed probability p_{ij} that it will change to state j. All those transformation probabilities p_{ij} form a transition matrix M_{r*r}.

Based on Definition 7, we have the definition of regular Markov chain as follows.

Definition 8. A Markov Chains is called a Regular Markov Chain, if and only if there exists an integer k, every element in M^k, where M is the transition matrix of the Markov chain, is positive.

Regular Markov chains have the following valuable property as follows.

Property 1 Grinstead and Snell, 1997. *If a Markov chain is a regular Markov chain and M is its transition matrix, we have:*

$$\exists M', \lim_{k\to\infty} M^k = M' \tag{9}$$

with

$$\forall i,j : m_{ji} = m_{1i} \tag{10}$$

and

$$\forall j : \sum_{1\leqslant i\leqslant r} m_{ji} = 1 \tag{11}$$

where m_{ij} is a element in M'.

Because all the rows of M' are the same, M' can be written as:

$$M' = \begin{pmatrix} m_1 & m_2 & \ldots & m_r \\ m_1 & m_2 & \ldots & m_r \\ \ldots & \ldots & \ldots & \ldots \\ m_1 & m_2 & \ldots & m_r \end{pmatrix} \tag{12}$$

with

$$\sum_{1\leqslant i\leqslant r} m_i = 1 \tag{13}$$

4.3.2. Convergence

Lemma 1. *A PAV model is a Markov chain.*

Proof. According to the definition of a PAV model, it is a Markov chain because the next activity of a viewer is only influenced by the current status. That is, the transform probability, $pr_{i\to j}$, from a vertex i to another vertex j, is fixed. □

Lemma 2. *A PAV model is a regular Markov chain.*

Proof. Based on formula (6), we can find out that every transition probability of a PAV model has a component, $\varepsilon/|N|$, and $\varepsilon/|N| > 0$ for $\varepsilon > 0$. Therefore, we have that $pr_{i\to j} > 0$ for any i,j ($\in N$). Let M be the transition matrix consisting of $pr_{i\to j}$. We know that each element of M is positive. According to Definition 8, a PAV model must be a regular Markov chain. □

Based on the above Lemma 2, we have the following conclusion.

Lemma 3. *Let M be a transition matrix of a PAV model and P be the vector of initial probability distribution. We have*

$$\lim_{k\to\infty} P(M^{k-1}) = (m_1, m_2, \ldots, m_r)$$

where $(m_1, m_2, \ldots, m_r)$ is the row vector of $\lim_{k\to\infty} M^k$.

Proof. First, we have

$$\lim_{k\to\infty} P(M^{k-1}) = P \lim_{k\to\infty} M^{k-1} \tag{14}$$

According to Property 1, we know

$$\lim_{k\to\infty} M^k = M'$$

and M' can be written as:

$$M' = \begin{pmatrix} m_1 & m_2 & \ldots & m_r \\ m_1 & m_2 & \ldots & m_r \\ \ldots & \ldots & \ldots & \ldots \\ m_1 & m_2 & \ldots & m_r \end{pmatrix} \tag{15}$$

Let $P = (p_1, p_2, \ldots, p_r)$. Based on formula (14) and (15), we have

$$\lim_{k\to\infty} P(M^{k-1}) = \left(\sum_{1\leqslant i\leqslant r} p_i \times m_1, \sum_{1\leqslant i\leqslant r} p_i \times m_2, \ldots \sum_{1\leqslant i\leqslant r} p_i \times m_r \right) \tag{16}$$

Because P be the vector of initial probability distribution, we have

$$\sum_{1\leqslant i\leqslant r} p_i = 1 \tag{17}$$

Therefore, we have

$$\lim_{k\to\infty} P(M^{k-1}) = (m_1, m_2, \ldots, m_r) \qquad \square$$

Lemma 3 shows that the result of Ranking_PAV will come to a stable distribution by running infinite times. In addition, Lemma 3 also indicates that the final probability distribution has nothing to do with the original probability distribution.

Obviously, we cannot run Ranking_PAV forever. Therefore, we stop running Ranking_PAV when the difference between the current probability distribution and the previous one is no more than a predefined threshold.

5. Experiments

For evaluating our model, we apply it to one real data set, and show its effectiveness by comparing it with some existing ranking system available on the Web. The existing ranking systems are three third-party ranking systems: Computer Science Conference Ranking[10] (CSRank), CiteSeer Rating[11] (CiteSeer) and ArnetMiner Rank.[12]

5.1. Data set

We use a real data set crawled from ACM Digital Library Portal.[13] The data set include all metadata of papers published by ACM from 1950 to 2008. After cleaning the data set, we obtain 196,044 authors, 162,256 papers and 636 conferences. Based on the above data set, we created a PAV graph.

It should be noted that DBLP, the widely used data set, is not used in our experiment. The reason is that DBLP does not contain the references of papers. Therefore, it is not suitable to our model.

5.2. Three exisiting ranking systems

CSRank contains an unofficial rank list for computer science conference.The ranking of conferences are objective and informal external source generated. The detailed procedure behind the ranking is unknown to the author. However, this ranking result is still a reference value. In CSRank, more than 700 international computer science conferences are separated into 11 areas or groups, including Databases, Applications & Media, Hardware & Architecture, Artificial Intelligence & Related Subjects, etc. Each area is separated into at most 4 echelons: Rank 1, Rank 2, Rank 3 and Unranked.

CiteSeer Rating is an impact factor rank automatically generated from documents in the CiteSeer database including approximately 500 journals and conferences. Each journal/conference's impact factor in a given year is calculated by formula $F = \log(N + 1)$, which N is the average citations of articles post on this journal/conference released or convened in this year. In this paper, we adopt CiteSeer 2003 and CiteSeer 2007. CiteSeer 2003 and CiteSeer 2007 are the ranking results published in 2003 and 2007 respectively.

ArnetMiner Rank is generated from ArnetMiner.net database by adopting impact factor as ranking methods. The calculation method of impact factor score is:

$$if(j,y) = \frac{\#citations_{y-1} + \#citations_{y-2}}{\#article_{y-1} + \#article_{y-2}} \tag{18}$$

where $if(j,y)$ represents the impact factor of journal (or conference) j in year y. $\#aritcile_{y-1}$ stands for the number of papers that were published by j in year $y-1$. $\#aritcile_{y-1}$ stands for the number of papers that were published by j in year $y-2$. $\#citations_{y-1}$ stands for the number of papers that cited papers published by j in year $y-1$. $\#citations_{y-2}$ stands for the number of papers that cited papers published by j in year $y-2$.

5.3. Conference ranking

In this section, we calculate the importance value of all conferences based on our method, and present the comparison of our result to CSRank, CiteSeer Rating and ArnetMiner Rank. In the following 3 tables, we consider several conferences according to three areas: Applications & Media, Databases and Hardware & Architecture. Column 2 to 6 denotes the importance value in our result, CSRank echelon, CiteSeer Rating in year 2003& 2007, ArnetMiner Rank of each conference respectively.

In these tables, we can see that our ranking results are consistent with those third-party rankings. In general, the more importance value in PAV model a conference is, the higher it ranks in CSRank and Arnet Rank and the more impact factor it gets in CiteSeer Rating. For instance, in Applications and Media area, SIGGRAPH, generally recognized as one of the most important and famous conference, get an importance value of 0.0188 in all computer science conferences, much higher than that of JCDL, a young conference, which has a history of only about 10 years.

Considering Applications and Media area in Table 4, the conferences, which get high importance value in our result, are also top-level international conferences at each sub-area of Applications and Media. For example, SIGGRAPH is one of the most influential conferences on Computer Graphics, so as SIGIR on Information Retrieval and ACM-MM on MultiMedia.

However, the CiteSeer Rating may contain errors on account of some mistake and missing data. Let's consider SIGGRAPH conference in Applications and Media area. There is a vacancy in CiteSeer Rating at year 2007.

Table 5 shows top three conferences generated by our method on the area of databases. We find that our ranking consists with CiteSeer 2007, CSRank, and Arnet Rank. To the best our knowledge, the ranking list of CiteSeer 2003 may not accord with the common view. Clearly, SIGMOD, started in 1975, is a premier international forum for database researchers. Compared with SIGMOD, KDD and CIKM, started in 1995 and 1992 respectively, seem to be younger conferences.

Table 6 shows five conferences generated by our method on the area of hardware and architectures. The result is a little complex. Our ranking result is the same as ArnetMiner Rank. The only difference between our ranking result and CSRank is that our method ranks DATE ahead of MICRO while CSRank think MICRO is better than DATE. In a whole, the difference between our method and CSRank is indistinctive. However, CiteSeer, no matter 2003 or 2007, seems obviously different from the other three ranking system. The reason is unclear because we do not know the details of CiteSeer. By consulting with some professionals of the area of hardware and architectures, we know that DAC, ISCA, ICCAD, DATE, and MICRO are all leading conferences.

For examining the effectiveness of the result generated by our method as a whole, we arrange our result in Table 7 to show the comparison of average importance value of conferences in CSRank. The last 2 columns denote the average importance value of conferences in 1st and 2nd echelon separately. From Table 7 we can see that our results are still consistent with CSRank. On the other hand,

[10] http://www3.ntu.edu.sg/home/assourav/crank.htm.
[11] http://CiteSeer.ist.psu.edu/stats/venues.
[12] http://www.arnetminer.net.
[13] http://portal.acm.org/portal.cfm.

Table 4
Conferences in applications and media area.

Conference	Importance value	CSRank echelon	CiteSeer 2007	CiteSeer 2003	ArnetMiner rank
SIGGRAPH	0.0188	Rank 1	–	0.49	5
SIGIR	0.0071	Rank 1	0.08	0.48	48
ACM-MM	0.0058	Rank 1	0.02	0.17	154
WWW	0.0033	Rank 1	0.09	0.98	22
SIGMETRICS	0.0028	Rank 1	0.09	1.41	67
JCDL	0.0021	Rank 2	0.02	0.08	305

Table 5
Conferences in databases area.

Conference	Importance value	CSRank echelon	CiteSeer 2007	CiteSeer 2003	ArnetMiner rank
SIGMOD	0.0091	Rank 1	0.90	0.12	8
KDD	0.0027	Rank 1	0.06	0.60	41
CIKM	0.0027	Rank 1	0.05	0.25	166

Table 6
Conferences in hardware and architecture area.

Conference	Importance value	CSRank echelon	CiteSeer 2007	CiteSeer 2003	ArnetMiner rank
DAC	0.0166	Rank 1	0.02	0.19	80
ISCA	0.0065	Rank 1	0.08	1.55	60
ICCAD	0.0049	Rank 1	0.01	0.13	196
DATE	0.0041	Rank 2	0	0.12	238
MICRO	0.0029	Rank 1	0.06	0.72	269

Table 7
The average importance value of conferences in different areas.

Area	1st echelon	2nd echelon
Applications and media	0.007551	0.006595
Artificial intelligence and related subjects	0.002715	0.002349
Hardware and architecture	0.007732	0.004121

for different areas, the more basic a subject is, the more influence it propagates. For example, Hardware and Architecture area is the basis of all computer subjects, so it has a much higher importance value than AI (Artificial Intelligence and Related Subjects).

5.4. Author ranking

For evaluating authors, we first rank all the authors by their importance value, and then classify the authors to the area where he mainly published his paper. Table 8 shows the Top-5 author ranked in database area. Column 2 is the rank of our model. Column 3 is the rank of ArnetMiner, which use H-index. Column 4 denotes the number of papers this author published querying from DBLP,[14] while column 5 denotes the number of papers this author published in SIGMOD conference.

Note that our rank is almost totally different from ArnetMiner. The reasons may lie in the following factors. First, the measure standards are different. Our method adopts probability distribution to measure the reputation of authors while ArnetMiner adopts H-index. Second, the bibliographic databases are different. The dataset used in this paper originates from ACM Digital Library while ArnetMiner use some online published bibliographic databases such as DBLP.

From the table we can see these Top-5 authors are all prestigious experts in Database area. They wrote many papers and a number of them were published by SIGMOD conference, which get the highest importance value in Database area by our method and other methods. Note that although the number of papers written by Surajit Chaudhuri is obviously least among the Top-5 authors, he is the author who wrote 41 papers published by SIGMOD. Compared with other authors, he wrote more papers published by SIGMOD. This greatly increases his important value in PAV model. Because H-index of an author is relevant to the number of papers that he wrote, it is understandable that the rank of Surajit Chaudhuri in ArnetMiner is much lower than his rank in our model.

Other authors we list here are also important experts in database area or its subarea, such as Prof. Christos Faloutsos and Jiawei Han in data mining and Serge Abiteboul in XML.

On the other hand, although the author Jiawei Han published many papers, his paper are mostly published in KDD, which is a younger conference compared to SIGMOD, and mainly influential in Data Mining, so Prof. Han does not rank highly in database area by our PAV model.

5.5. Paper ranking

We consider all papers, calculate their importance value and sort them. We show Top-5 papers published in the area of database and information retrieval by Tables 9 and 10. The 2nd to 4th columns denote the importance value of paper, the venue and year it published and the number of paper's citations from Google scholar search engine by 11 Feb. 2011.

From Table 9 we can see most important papers in database area are published in SIGMOD conference and cited thousands of times. For instance, "Mining association rules between sets of items in large databases" first introduced the problem of mining association rules in large databases and proposed the famous "Apriori" algorithm to solve it. Many follow-up researches are carried out around this paper and form an important branch in data mining discipline. Other papers such as "The R*-tree: an efficient and robust access method for points and rectangles" and "Direct spatial search on pictorial databases using packed R-trees" pro-

[14] http://www.informatik.uni-trier.de/~ley/db/index.html.

Table 8
Top-5 author ranked in databases.

Author	Rank by our PAV model	Rank by ArnetMiner (H-index)	Num. in DBLP	Num. in SIGMOD
Surajit Chaudhuri	1	41	177	41
Christos Faloutsos	2	11	289	19
Serge Abiteboul	3	7	238	24
Hector Garcia-Molina	4	1	378	30
Jiawei Han	5	2	420	28

Table 9
Top-5 papers ranked in database area.

Title of paper	Importance value (10^{-4})	Publish	Citations from Google scholar
Mining association rules between sets of items in large databases	1.643	SIGMOD'93	8982
The R*-tree: an efficient and robust access method for points and rectangles	1.034	SIGMOD'90	3438
SEQUEL:A structured English query language	0.611	SIGMOD'74	390
Implementing data cubes efficiently	0.469	SIGMOD'96	1294
Direct spatial search on pictorial databases using packed R-trees	0.459	SIGMOD'95	380

Table 10
Top-5 papers ranked in information retrieval area.

Title of paper	Importance value (10^{-5})	Publish	Citations from Google scholar
Scatter/Gather: a cluster-based approach to browsing large document collections	8.464	SIGIR'92	1426
A language modeling approach to information retrieval	8.182	SIGIR'98	1433
The information visualizer, an information workspace	7.048	CHI'91	537
Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval	5.880	SIGIR'94	685
Improved algorithms for topic distillation in a hyperlinked environment	5.237	SIGIR'98	881

posed R-Tree and R*-Tree, two very important data structures, to solve the problem of index structure for multidimensional data.

In Table 10, we can see that most important papers are published in SIGIR, a top-level international conference in this area. Many well-known papers are included. For example, "A language modeling approach to information retrieval" proposed a language model, a probability model to describe the words distribution of language. This model is widely used in speech recognition, natural language processing, machine translation and other fields.

It should note that because our model ranking paper without considering the date when these papers published. Generally speaking, the earlier the paper was published, the more other papers may cite it. Therefore, these Top-5 papers are all published before 2000. However, it is easy to extend our model to contain date-impact. For example, we can introduce time-orient decay factor to PAV model or just compare papers that were published in the same period of time.

6. Conclusions

In this paper, we proposed a novel model, called PAV, for ranking heterogeneous objects, such as authors, papers, and venues in bibliographic information networks. According to this model, a bibliographic information network is represented by a weight directed graph, where a vertex stands for an object, an edge stands for the link between objects, and a weight over an edge stands for the contribution that one object devotes to the reputation or importance of the corresponding object sharing the same edge. Based on PAV model, we transform the problem of ranking objects into the problem of estimating probability parameters. For estimating probability, we presented an algorithm based on matrix computing. Specially, we showed our algorithm could be running efficiently by proofing that the underlying computing method is convergent. Our experiments on a real data set crawled from ACM Digital Library show that the PAV model is effective.

In future, we will extensively study the performance of PAV model on other real bibliographic databases. Another interesting work is to extend the PAV model to other mining tasks, such as clustering and classification.

Acknowledgement

This work is partially supported by Project 61170091 supported by National Natural Science Foundation of China and Project 2009AA01Z136 supported by the National High Technology Research and Development Program of China (863 Program).

References

Balmin, A., Hristidis, V., & Papakonstantinou. Y. (2004). ObjectRank: Authority-Based Keyword Search in Databases. In *VLDB*.

Bensman, S. J., & Wilder, S. J. (1998). Scientific and technical serials holdings optimization in an inefficient market: A LSU serials redesign project exercise. *Library Resources and Technical Services, 42*(3), 147–242.

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. In WWW.

Burrell, Q. L. (2007). Hirsch index or Hirsch rate? Some thoughts arising from Liang's data. *Scientometrics, 73*(1), 19–28.

Egghe, L. (2006a). An improvement of the h-index: The g-index. *ISSI Newsletter, 2*, 8–9.

Egghe, L. (2006b). Theory and practice of the g-index. *Scientometrics, 69*, 131–152.

Frandsen, T. F., & Rousseau, R. (2005). Article impact calculated over arbitrary periods. *Journal of the American Society for Information Science and Technology, 56*(1), 58–62.

Garfield, E. (1972). Citation analysis as a tool in journal evaluation. *Science, 178*(4060), 471–479.

Garfield, E. (1998). Long-term vs. short-term journal impact: Does it matter? *The Scientist, 12*(3), 11–12.

Grinstead, C., & Snell, J. L. (1997). *Introduction to probability* (2nd ed.). Providence, RI: American Mathematical Society.

Hirsch, J. E. (2005). An index to quantify an individual's scientific research output. *Proceedings of the National Academy of Sciences, 102*(46), 16569–16572.

Hwang, H., Hristidis, V., & Papakonstantinou, Y. (2006). ObjectRank: A system for authority-based search on databases. In *SIGMOD*.

James, H. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica, 57*(2), 357–384.

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

Manning, C. D., Raghavan, P., & Schutze, H. (2008). *Introduction to information retrieval*. Cambridge University Press.

Peter, K., David, L., & Eugene, S. (2009). FOG: Fragment Optimized Growth Algorithm for the de Novo Generation of Molecules occupying Druglike Chemical. *Journal of Chemical Information and Modeling, 49*(7), 1630–1642.

Sun, Y., Wu, T., Cheng, H., Han, J., Yin, X., & Zhao, P. (2008). BibNetMiner: Mining Bibliographic Information Networks. In *SIGMOD*.

Sun, Y., Han, J., Gao, J., & Yu, Y. (2009). iTopicModel: Information network-integrated topic modeling. In *ICDM*.

Sun, Y., Yu, Y., & Han, J. (2009). Ranking-based clustering of heterogeneous information networks with star network schema. In *KDD*.

Sun, Y., Han, J., Zhao, P., Yin, Z., Cheng, H., & Wu, T. (2009). RankClus: Integrating Clustering with Ranking for Heterogeneous Information Network Analysis. In *EDBT*.

Tang, J., Jin, R., & Zhang. J. (2008). A topic modeling approach and its integration into the random walk framework for academic search. In *ICDM*.

Tang, J., Zhang, D., & Yao, L. (2007). Social network extraction of academic researchers. In *ICDM*.

Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z. (2008). ArnetMiner: Extraction and mining of academic social networks. In *KDD*.

Trivedi, K. S. (2002). *Probability and statistics with reliability, queueing, and computer science applications*. Inc. New York: John Wiley & Sons.

Wang, C., Han, J., Jia, Y., Tang, J., Zhang, D., Yu, Y., & Guo, J. (2010). Mining advisor–advisee relationships from research publication networks. In *KDD*.

Yin, X., Han, J., Yu, P. S. (2006). Linkclus: Efficient clustering via heterogeneous semantic links. In *VLDB*.