Chapter 4
Path-Based Ranking and Clustering

Abstract As newly emerging network models, heterogeneous information networks have many unique features, e.g., complex structures and rich semantics. Moreover, meta path, the sequence of relations connecting two object types, is an effective tool to integrate different types of objects and mine the semantic information in this kind of networks. The unique characteristics of meta path make the data mining on heterogeneous network more interesting and challenging. In this chapter, we will introduce two basic data mining tasks, ranking and clustering, on heterogeneous information network. Furthermore, we introduce the HRank method to evaluate the importance of multiple types of objects and meta paths, and present the HeProjI algorithm to solve the heterogeneous network projection and integration of clustering and ranking tasks.

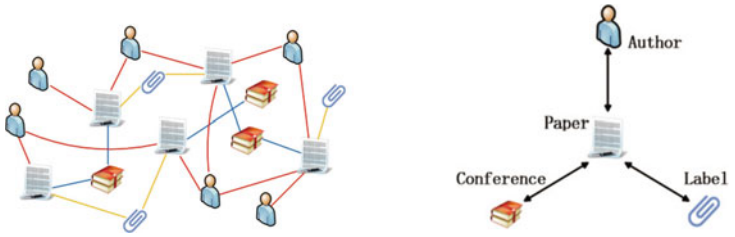
4.1 Meta Path-Based Ranking

4.1.1 Overview

It is an important research problem to evaluate object importance or popularity, which can be used in many data mining tasks. Many methods have been developed to evaluate object importance, such as PageRank [13], HITS [7], and SimRank [5]. In these literatures, objects ranking is done in a homogeneous network in which objects or relations are the same. For example, both PageRank and HITS rank the web pages in WWW.

However, in many real network data, there are many different types of objects and relations, which can be organized as heterogeneous networks. Formally, heterogeneous information networks (HIN) are the logical networks involving multiple types of objects as well as multiple types of links denoting different relations [4]. Recently, many data mining tasks have been exploited in this kind of networks,

C. Shi and P.S. Yu, *Heterogeneous Information Network Analysis and Applications*, Data Analytics, DOI 10.1007/978-3-319-56212-4_4



(a) Heterogeneous network (b) Network schema

Fig. 4.1 A heterogeneous information network example on bibliographic data. **a** shows heterogeneous objects and their relations. **b** shows the network schema

such as similarity measure [14, 25], clustering [23], and classification [6], among which ranking is an important but seldom exploited task.

Figure 4.1a shows an HIN example in bibliographic data, and Fig. 4.1b illustrates its network schema which depicts object types and their relations. In this example, it contains objects from four types of objects: papers (P), authors (A), labels (L, categories of papers), and conferences (C). There are links connecting different types of objects. The link types are defined by the relation between the two object types. In this network, several interesting, yet seldom exploited, ranking problems can be proposed.

- One may be interested in the importance of one type of objects and ask the following questions:
  Q. 1.1 Who are the most influential authors?
  Q. 1.2 Who are the most influential authors in data mining field?
- As we know, some object types have an effect on each other. For example, influential authors usually publish papers in reputable conferences. So one may pay attention to the importance of multiple types of objects simultaneously and ask the following questions:
  Q. 2.1 Who are the most influential authors and which reputable conferences did those influential authors publish their papers on?
  Q. 2.2 Who are the most influential authors and which reputable conferences did those influential authors publish their papers on in data mining field?
- Furthermore, one may wonder which factor mostly affects the importance of objects, since the importance of objects is affected by many factors. So he may ask the questions like this:
  Q. 3 Who are the most influential authors and which factors make those most influential authors be most influential?

Although the ranking problem in homogeneous networks has been well studied, the above ranking problems are unique in HIN (especially *Q. 2* and *Q. 3*), which are seldom studied until now. Since there are multiple types of objects in HIN, it is possible to analyze the importance of multiple types of objects (i.e., *Q. 2*) as well as affecting factors (i.e., *Q. 3*) together.

In this chapter, we study the ranking problem in HIN and propose a ranking method, HRank, to evaluate the importance of multiple types of objects and meta paths in HIN. For *Q. 1* and *Q. 2*, a path-based random walk model is proposed to evaluate the importance of single or multiple types of objects. The different meta paths connecting two types (same or different types) of objects have different semantics and transitive probability, and thus lead to different random walk processes and ranking results. Although meta path has been widely used to capture the semantics in HIN [14, 25], it coarsely depicts object relations. By employing the meta path, we can answer the *Q. 1.1* and *Q. 2.1*, but cannot answer the *Q. 1.2* and *Q. 2.2*. In order to overcome the shortcoming existing in meta path, we propose the *constrained meta path* concept, which can effectively describe this kind of subtle semantics. The constrained meta path assigns constraint conditions on meta path. Through adopting the constrained meta path, we can answer the *Q. 1.2* and *Q. 2.2*.

Moreover, in HIN, based on different paths, the objects have different ranking values. The comprehensive importance of objects should consider all kinds of factors (the factors can be embodied by constrained meta paths), which have different contribution to the importance of objects. In order to evaluate the importance of objects and meta paths simultaneously (i.e., answer *Q. 3*), we further propose a co-ranking method which organizes the relation matrices of objects on different constrained meta paths as a tensor. A random walk process is designed on this tensor to co-rank the importance of objects and paths simultaneously. That is, random walkers surf in the tensor, where the stationary visiting probability of objects and meta paths is considered as the HRank score of objects and paths.

4.1.2 The HRank Method

Since the importance of objects is related to the meta path designated by users, we propose the path-based ranking method HRank in heterogeneous networks. In order to answer the three ranking problems proposed above, we design three versions of HRank, respectively.

4.1.2.1 Constrained Meta Path

As an effective semantic capturing method, the meta path has been widely used in many data mining tasks in HIN, such as similarity measure [14, 25], clustering [23], and classification [8]. However, meta path may fail to capture subtle semantics in some situations. Taking Fig. 4.1b as an example, the *APA* path cannot reveal the co-author relations in a certain research field, such as data mining and information retrieval. Although Jiawei Han has co-worked many papers with Philip S. Yu in the data mining field, they never co-work in the operation system field. The *APA* path cannot subtly reflect this difference.

In order to overcome the shortcomings in meta path, we propose the concept of constrained meta path, defined as follows.

Definition 4.1 (*Constrained meta path*) A constrained meta path is a meta path based on a certain constraint which is denoted as $\mathtt{CP} = P|\mathtt{C}$. $P = (A_1 A_2 \ldots A_l)$ is a meta path, while C represents the constraint on the objects in the meta path.

Note that the C can be one or multiple constraint conditions on objects. Taking Fig. 4.1b as an example, the constrained meta path $APA|P.L =$ "*DM*" represents the co-author relations of authors in data mining field through constraining the label of papers with data mining (DM). Similarly, the constrained meta path $APCPA|P.L =$ "*DM*"&&$C =$ "*CIKM*" represents the co-author relations of authors in CIKM conference, and the papers of authors are in data mining field. Obviously, compared to meta path, the constrained meta path conveys richer semantics by subdividing meta paths under distinct conditions. Particularly, when the length of meta path is 1 (i.e., a relation), the constrained meta path degrades to a **constrained relation**. In other words, the constrained relation confines constraint conditions on objects of the relation.

For a relation $A \xrightarrow{R} B$, we can obtain its transition probability matrix as follows.

Definition 4.2 (*Transition probability matrix*) W_{AB} is an adjacent matrix between type A and B on relation $A \xrightarrow{R} B$. U_{AB} is the normalized matrix of W_{AB} along the row vector, which is the transition probability matrix of $A \xrightarrow{R} B$.

Then, we make some constraints on objects of the relation $A \xrightarrow{R} B$ (i.e., constrained relation). We can have the following definition.

Definition 4.3 (*Constrained transition probability matrix*) W_{AB} is an adjacent matrix between type A and B on relation $A \xrightarrow{R} B$. Suppose there is a constraint C on object type A. The constrained transition probability matrix U'_{AB} of constrained relation $R|\mathtt{C}$ is $U'_{AB} = M_{\mathtt{C}} U_{AB}$, where $M_{\mathtt{C}}$ is the constraint matrix generated by the constraint condition C on object type A.

The constraint matrix $M_{\mathtt{C}}$ is usually a diagonal matrix whose dimension is the number of objects in object type A. The element in the diagonal is 1 if the corresponding object satisfies the constraint, else the element in the diagonal is 0. For example, in the path $PC|C =$ "*CIKM*", $M_{\mathtt{C}}$ is a diagonal matrix of conferences, where the "*CIKM*" column is 1 and the others are 0. Similarly, we can confine the constraint on the object type B or both types. Note that the transition probability matrix is a special case of the constrained transition probability matrix, when we let the constraint matrix $M_{\mathtt{C}}$ be the identity matrix I.

Given a network $G = (V, E)$ following a network schema $S = (\mathtt{A}, \mathtt{R})$, we can define the meta path-based reachable probability matrix as follows.

Definition 4.4 (*Meta path-based reachable probability matrix)* For a meta path $P = (A_1A_2 \cdots A_{l+1})$, the meta path-based reachable probability matrix PM is defined as $PM_P = U_{A_1A_2}U_{A_2A_3} \cdots U_{A_lA_{l+1}}$. $PM_P(i, j)$ represents the probability of object $i \in A_1$ reaching object $j \in A_{l+1}$ under the path P.

Similarly, we have the following definition for constrained meta path.

Definition 4.5 (*Constrained meta path-based reachable probability matrix*) For a constrained meta path $\mathtt{CP} = (A_1A_2 \cdots A_{l+1}|\mathtt{C})$, the constrained meta path-based reachable probability matrix is defined as $PM_{\mathtt{CP}} = U'_{A_1A_2}U'_{A_2A_3} \cdots U'_{A_lA_{l+1}}$. $PM_{\mathtt{CP}}(i, j)$ represents the probability of object $i \in A_1$ reaching object $j \in A_{l+1}$ under the constrained meta path $\mathtt{P}|\mathtt{C}$.

In fact, if there is no constraint on the objects of a relation $A_i \xrightarrow{R} A_{i+1}$, $U'_{A_iA_{i+1}}$ is equal to $U_{A_iA_{i+1}}$. If there is a constraint on the objects, we only consider the objects that satisfy the constraint. For simplicity, we use the reachable probability matrix and the M_P to represent the constrained meta path-based reachable probability matrix in the following section.

4.1.2.2 Ranking Based on Symmetric Meta Paths

In order to evaluate the importance of one type of objects (i.e., *Q. 1*), we design the HRank-SY method based on symmetric constrained meta paths, since the constrained meta paths connecting one type of objects are usually symmetric, such as $APA|P.L =$ "*DM*".

For a symmetric constrained meta path $P = (A_1A_2 \ldots A_l|C)$, P is equal to P^{-1} and A_1 and A_l are the same. Similar to PageRank [13], the importance evaluation of object A_1 (i.e., A_l) can be considered as a random walk process in which random walkers wander from type A_1 to type A_l along the path P. The HRank value of object A_1 (i.e., $R(A_1|P)$) is the stable visiting probability of random walkers, which is defined as follows:

$$R(A_1|P) = \alpha R(A_1|P)M_P + (1 - \alpha)E \tag{4.1}$$

where M_P is the constrained meta path-based reachable probability matrix as defined above. E is the restart probability vector for convergence. It is set equally for all objects of type A_1, which is $1/|A_1|$. α is the decay factor, which can be set with 0.85 as the parameter experiments suggested. HRank-SY and PageRank both have the same idea that the importance of objects is decided by the visiting probability of random surfers. Different from PageRank, the random surfers in HRank-SY should wander along the constrained meta path to visit objects.

As shown in Fig. 4.2, the red broken line illustrates an example of the process of calculating rank values, where the CP is $APA|P.L =$ "*DM*". The concrete calculating process is as follows:

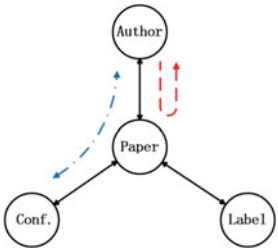


Fig. 4.2 An example of the computation process of HRank. The *blue* and *red broken lines* represent the process on the symmetric and asymmetric constrained meta path, respectively

$$\begin{aligned} R(Author|\texttt{CP}) &= \alpha R(Author|\texttt{CP})M_{\texttt{CP}} + (1-\alpha)E \\ M_{\texttt{CP}} &= U'_{AP}U'_{PA} = U_{AP}M_PM_PU_{PA} \end{aligned} \tag{4.2}$$

where M_P is the constraint matrix on object type P (paper).

4.1.2.3 Ranking Based on Asymmetric Meta Paths

For the question *Q. 2*, we propose the HRank-AS method based on asymmetric constrained meta paths, since the paths connecting different types of objects are asymmetric. For an asymmetric constrained meta path $P = (A_1A_2 \ldots A_l|\texttt{C})$, P is not equal to P^{-1}. Note that A_1 and A_l are either of the same or different types, such as $APC|P.L =$ "*DM*" and $PCPLP|C =$ "*CIKM*".

Similarly, HRank-AS is also based on a random walk process that random walkers wander between A_1 and A_l along the path. The ranks of A_1 and A_l can be seen as the visiting probability of walkers, which are defined as follows:

$$\begin{aligned} R(A_l|P^{-1}) &= \alpha R(A_1|P)M_P + (1-\alpha)E_{A_l} \\ R(A_1|P) &= \alpha R(A_l|P^{-1})M_{P^{-1}} + (1-\alpha)E_{A_1} \end{aligned} \tag{4.3}$$

where M_P and $M_{P^{-1}}$ are the reachable probability matrix of path P and P^{-1}. E_{A_1} and E_{A_l} are the restart probability of A_1 and A_l. Obviously, HRank-SY is the special case of HRank-AS. When the path P is symmetric, Eq. 4.3 is the same with Eq. 4.1.

The blue broken line in Fig. 4.2 illustrates an example which simultaneously evaluates the importance of authors and conferences. Here, the CP is $APC|P.L =$ "*DM*". The concrete calculating process is as follows:

$$\begin{aligned} R(Conf.|\texttt{CP}) &= \alpha R(Aut.|\texttt{CP})M_{\texttt{CP}} + (1-\alpha)E_{Conf.} \\ R(Aut.|\texttt{CP}) &= \alpha R(Conf.|\texttt{CP})M_{\texttt{CP}^{-1}} + (1-\alpha)E_{Aut.} \\ M_{\texttt{CP}} &= U'_{AP}U'_{PC} = U_{AP}M_PM_PU_{PC} \\ M_{\texttt{CP}^{-1}} &= U'_{CP}U'_{PA} = U_{CP}M_PM_PU_{PA} \end{aligned} \tag{4.4}$$

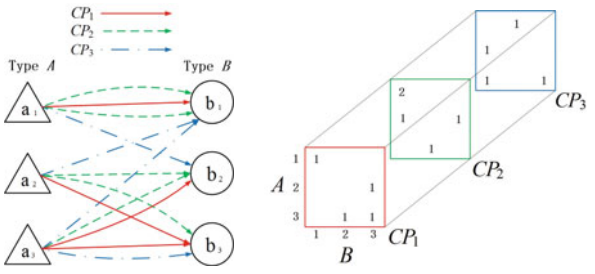
where M_P is the constraint matrix on object type P (paper).

4.1.2.4 Co-ranking for Objects and Relations

Until now, we have created methods to rank same or different types of objects under a certain constrained meta path. However, there are many constrained meta paths in heterogeneous networks. It is an important issue to automatically determine the importance of paths [23, 25], since it is usually hard for us to identify which relation is more important in real applications. To solve this problem (i.e., *Q. 3*), we propose the HRank-CO to co-rank the importance of objects and relations. The basic idea is based on an intuition that important objects are connected to many other objects through a number of important relations and important relations connect many important objects. So we organize the multiple relation networks with a tensor, and a random walk process is designed on this tensor. The method not only can comprehensively evaluate the importance of objects by considering all constrained meta paths, but also can rank the contribution of different constrained meta paths.

In Fig. 4.3a, we show an example of multiple relations among objects, generated by multiple meta paths. There are three objects of type A, three objects of type B, and three types of relations among them. These relations are generated by three constrained meta paths with type A as the source type and type B as the target type. To describe the multiple relations among objects, we use the representation of tensor which is a multidimensional array. We call $X = (x_{i,j,k})$ a third-order tensor, where $x_{i,j,k} \in R$, for $i = 1, \cdots, m$, $j = 1, \cdots, l$, $k = 1, \cdots, n$. $x_{i,j,k}$ represents the times that object i is related to object k through the jth constrained meta path. For example, Fig. 4.3b is a three-way array, where each two-dimensional slice represents an adjacency matrix for a single relation. So the data can be represented as a tensor of size $3 \times 3 \times 3$. In the multirelational network, we define the transition probability tensor to present the transition probability among objects and relations.

Definition 4.6 (*Transition probability tensor*) In a multirelational network, X is the tensor representing the network. F is the normalized tensor of X along the column vector. R is the normalized tensor of X along the tube vector. T is the normalized



(a) Multiple relations (b) Tensor representation

Fig. 4.3 An example of multirelations of objects generated by multiple paths: **a** the graph representation; **b** the corresponding tensor representation

tensor of X along the row vector. F, R, and T are called the transition probability tensors which can be denoted as follows:

$$\begin{aligned} f_{i,j,k} &= \frac{x_{i,j,k}}{\sum_{i=1}^{m} x_{i,j,k}} \quad i = 1, 2, \ldots, m \\ r_{i,j,k} &= \frac{x_{i,j,k}}{\sum_{j=1}^{l} x_{i,j,k}} \quad j = 1, 2, \ldots, l \\ t_{i,j,k} &= \frac{x_{i,j,k}}{\sum_{k=1}^{n} x_{i,j,k}} \quad k = 1, 2, \ldots, n \end{aligned} \tag{4.5}$$

$f_{i,j,k}$ can be interpreted as the probability of object i (of type A) being the visiting object when relation j is used and the current object being visited is object k (of type B), $r_{i,j,k}$ represents the probability of using relation j given that object k is visited from object i, and $t_{i,j,k}$ can be interpreted as the probability of object k being visited, given that object i is currently the visiting object and relation j is used. The meaning of these three tensors can be defined formally as follows:

$$\begin{aligned} f_{i,j,k} &= Prob(X_t = i | Y_t = j, Z_t = k) \\ r_{i,j,k} &= Prob(Y_t = j | X_t = i, Z_t = k) \\ t_{i,j,k} &= Prob(Z_t = k | X_t = i, Y_t = j) \end{aligned} \tag{4.6}$$

in which X_t, Z_t, and Y_t are three random variables representing visiting at certain object of type A or type B and using certain relation respectively at the time t.

Now, we define the stationary distributions of objects and relations as follows:

$$\begin{aligned} x &= (x_1, x_2, \cdots, x_m)^T \\ y &= (y_1, y_2, \cdots, y_l)^T \\ z &= (z_1, z_2, \cdots, z_n)^T \end{aligned} \tag{4.7}$$

in which

$$\begin{aligned} x_i &= \lim_{t \to \infty} Prob(X_t = i) \\ y_j &= \lim_{t \to \infty} Prob(Y_t = j) \\ z_k &= \lim_{t \to \infty} Prob(Z_t = k) \end{aligned} \tag{4.8}$$

From the above equations, we can get:

$$\begin{aligned} Prob(X_t = i) &= \sum_{j=1}^{l}\sum_{k=1}^{n} f_{i,j,k} \times Prob(Y_t = j, Z_t = k) \\ Prob(Y_t = j) &= \sum_{i=1}^{m}\sum_{k=1}^{n} r_{i,j,k} \times Prob(X_t = i, Z_t = k) \\ Prob(Z_t = k) &= \sum_{i=1}^{m}\sum_{j=1}^{l} t_{i,j,k} \times Prob(X_t = i, Y_t = j) \end{aligned} \tag{4.9}$$

where $Prob(Y_t = j, Z_t = k)$ is the joint probability distribution of Y_t and Z_t, $Prob(X_t = i, Z_t = k)$ is the joint probability distribution of X_t and Z_t, and $Prob(X_t = i, Y_t = j)$ is the joint probability distribution of X_t and Y_t.

To obtain x_i, y_j, and z_k, we assume that X_t, Y_t, and Z_t are all independent from each other which can be denoted as below:

$$\begin{aligned} Prob(X_t = i, Y_t = j) &= Prob(X_t = i)Prob(Y_t = j) \\ Prob(X_t = i, Z_t = k) &= Prob(X_t = i)Prob(Z_t = k) \\ Prob(Y_t = j, Z_t = k) &= Prob(Y_t = j)Prob(Z_t = k) \end{aligned} \tag{4.10}$$

Consequently, through combining the equations with the assumptions above, we get:

$$\begin{aligned} x_i &= \sum_{j=1}^{l}\sum_{k=1}^{n} f_{i,j,k} y_j z_k, i = 1, 2, \ldots, m, \\ y_j &= \sum_{i=1}^{m}\sum_{k=1}^{n} r_{i,j,k} x_i z_k, j = 1, 2, \ldots, l, \\ z_k &= \sum_{i=1}^{m}\sum_{j=1}^{l} t_{i,j,k} x_i y_j, k = 1, 2, \ldots, n. \end{aligned} \tag{4.11}$$

The equations above can be written in a tensor format:

$$x = Fyz, y = Rxz, z = Txy \tag{4.12}$$

with $\sum_{i=1}^{m} x_i = 1$, $\sum_{j=1}^{l} y_j = 1$, and $\sum_{k=1}^{n} z_k = 1$.

According to the analysis above, we can design the following algorithm to co-rank the importance of objects and relations.

Algorithm 4.1 HRank-CO Algorithm

Input:
Three tensors F, T and R, three initial probability distributions x_0, y_0 and z_0 and the tolerance ε.
Output:
Three stationary probability distributions x, y and z.
Procedure:
Set $t = 1$;
repeat
  Compute $x_t = Fy_{t-1}z_{t-1}$;
  Compute $y_t = Rx_tz_{t-1}$;
  Compute $z_t = Tx_ty_t$;
until $||x_t - x_{t-1}|| + ||y_t - y_{t-1}|| + ||z_t - z_{t-1}|| < \varepsilon$

4.1.3 Experiments

In this section, we do experiments to validate the effectiveness of three versions of HRank on three real datasets, respectively. Here we use three real datasets: DBLP dataset [14, 25], ACM dataset [14], and IMDB dataset [16].

4.1.3.1 Ranking of Homogeneous Objects

Since the homogeneous objects are connected by symmetric constrained meta paths, the experiments validate the effectiveness of HRank-SY on symmetric constrained meta paths.

Experiment Study on Symmetric Constrained Meta Paths This experiment ranks the same type of objects by designating a symmetric constrained meta path on ACM dataset. Here, we rank the importance of authors through the symmetric meta path APA, which considers the co-author relations among authors. We also employ two constrained meta paths $APA|P.L =$ "$H.2$" and $APA|P.L =$ "$H.3$", where the categories of ACM $H.2$ and $H.3$ represent "database management" and "information storage/retrieval," respectively. That is, two constrained meta paths subtly consider the co-author relations in database/data mining field and information retrieval field, respectively. We employ HRank-SY to rank the importance of authors based on these three paths. As the baseline methods, we rank the importance of authors with PageRank and the degree of authors (called Degree method). We directly run PageRank on the whole ACM network by ignoring the heterogeneity of objects. Since the results of PageRank mix all types of objects, we select the author type from the ranking list as the final results.

The top ten authors of each method are shown in Table 4.1. We can find that all these ranking lists have some common influential authors except that of PageRank. The results of PageRank include some not very well-known authors in database/information retrieval (DB/IR) field, such as Ming Li and Wei Wei, although they may be very influential in other fields. We know that the PageRank values of

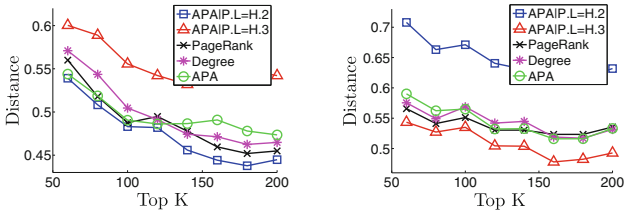
Table 4.1 Top ten authors of different methods on ACM dataset. The number in the parenthesis of the fifth column means the rank of authors in the whole ranking list returned by PageRank

Rank	*APA*	$APA\|P.L =$ "$H.3$"	$APA\|P.L =$ "$H.2$"	PageRank	Degree
1	Jiawei Han	W. Bruce Croft	Jiawei Han	Ming Li(1522)	Jiawei Han
2	Philip Yu	ChengXiang Zhai	Christos Faloutsos	Wei Wei(2072)	Philip Yu
3	Christos Faloutsos	James Allan	Philip Yu	Jiawei Han(5385)	ChengXiang Zhai
4	Zheng Chen	Jamie Callan	Jian Pei	Tao Li(6090)	Zheng Chen
5	Wei-Ying Ma	Zheng Chen	H. Garcia-Molina	Hong-Jiang Zhang(6319)	Christos Faloutsos
6	ChengXiang Zhai	Ryen W. White	Jeffrey F. Naughton	Wei Ding(6354)	Ravi Kumar
7	W. Bruce Croft	Wei-Ying Ma	Divesh Srivastava	Jiangong Zhang(7285)	W. Bruce Croft
8	Scott Shenker	Jian-Yun Nie	Raghu Ramakrishnan	Christos Faloutsos(7895)	Wei-Ying Ma
9	H. Garcia-Molina	Gerhard Weikum	Charu C. Aggarwal	Feng Pan(8262)	Gerhard Weikum
10	Ravi Kumar	C. Lee Giles	Surajit Chaudhuri	Hongyan Liu(8440)	Divesh Srivastava

objects are decided by their degrees to a large extent, so the rank values of affiliation objects are high due to their high degrees. It improves the rank values of author objects connecting multiple high-ranking affiliations. The bad results of PageRank show that the ranking in heterogeneous networks should consider the heterogeneity of objects. Otherwise, it cannot distinguish the effect of different types of links. Moreover, we can also observe that the results of HRank with constrained meta paths have obvious bias on the field it assigns. For example, the path $APA|P.L =$ "$H.3$" reveals the important authors in information retrieval field, such as W. Bruce Croft, ChengXiang Zhai, and James Allan. However, the path $APA|P.L =$ "$H.2$" returns the influential authors in database and data mining field, such as Jiawei Han and Christos Faloutsos. For the meta path *APA*, it mingles well-known authors in these two fields. The results illustrate that the constrained meta paths are able to capture subtle semantics by deeply disclosing the most influential authors in a certain field.

Quantitative Comparison Experiments Based on the results returned by five methods, we can obtain five candidate ranking lists of authors in ACM dataset. To evaluate the results quantitatively, we crawled data as ground truth from two well-known websites. The first ground truth provides the author ranks from Microsoft Academic Search.[1] Specifically, we crawled two standard ranking lists of authors in two

[1] http://academic.research.microsoft.com/.



(a) DB field (b) IR field

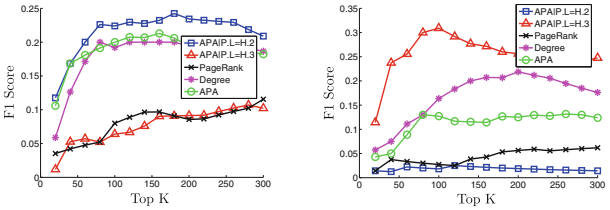
Fig. 4.4 The distances between the ranking lists obtained by different methods and the standard ranking lists on different fields on ACM dataset. The ground truth is from Microsoft Academic Search

academic fields: DB and IR. Then, we compare the difference between our candidate ranking lists and the standard ranking lists. In order to measure the quality of the ranking results, we use the *Distance* criterion proposed in [12], which is defined as follows.

$$D(R, R') = \frac{\sum_{i=1}^{n}[(n-i) \times \sum_{j=1 \wedge R'_j \notin \{R_1,\ldots,R_i\}}^{i} 1]}{\sum_{i=1}^{\lfloor \frac{n}{2} \rfloor}[(n-i) \times i] + \sum_{i=\lfloor \frac{n}{2} \rfloor+1}^{n}[(n-i) \times (n-i)]} \tag{4.13}$$

where R_i represents the ith object in ranking list R, while R'_j denotes the jth object in ranking list R'. And n is the total number of objects in the ranking lists. Note that the numerator of the formula measures the real distance between the two rankings, and the denominator of the formula is used to normalize the real distance to a number between 0 and 1. So the criterion not only measures the number of mismatches between these two lists, but also considers the position of these mismatches. The smaller *Distance* means the smaller difference (i.e., better performance).

In this experiment, we compare the five candidate ranking lists with each of the two standard ranking lists from Microsoft Academic Search and the *Distance* results are shown in Fig. 4.4. We can observe an obvious phenomenon: The results obtained by the constrained meta paths have the smallest *Distance* on its corresponding field, while they have the largest *Distance* on other fields. For example, HRank with the path $APA|P.L =$ "$H.2$" has the smallest *Distance* on the DB field in Fig. 4.4a, while it has the largest *Distance* on IR field in Fig. 4.4b. The reason lies in that the path $APA|P.L =$ "$H.2$" focuses on the authors in the DB field. Meanwhile, these authors deviate from those in the IR field. The results further illustrate that the constrained meta path can disclose the influential authors in a certain field more correctly. Since the meta path (i.e., *APA*) considers the co-author relationship on all fields, it achieves mediocre performances on these two fields. In fact, the HRank with meta path *APA* only achieves closer performances to PageRank and Degree methods. It implies that the constrained meta path in HRank indeed helps to improve the ranking performances in a specific field.



(a) DB field (b) IR field

Fig. 4.5 F1 accuracy of the ranking lists obtained by different methods on different fields on ACM dataset. The ground truth is from ArnetMiner

Furthermore, we quantitatively evaluate the results according to the second ground truth from ArnetMiner [26] that offers comprehensive search and mining services for academic community.[2] Specifically, we crawl the first 200 authors as experts in DB and IR fields through searching "data mining" and "information retrieval." Since these 200 experts have no ranking order, we evaluate the accuracy of the top k authors of five candidate ranking lists with the F1 score. From the results shown in Fig. 4.5, we can observe the same phenomena. That is, the constrained meta paths always achieve the best performances on their corresponding fields, while they have the worst performances on other fields (note that the higher F1 score means the better performances). Moreover, the meta paths also have the moderate performances. The experiments on both ground truths confirm that HRank is able to improve the ranking performances in a specific field through assigning constrained meta paths.

4.1.3.2 Ranking of Heterogeneous Objects

Then, the experiments validate the effectiveness of HRank-AS on asymmetric constrained meta paths.

Experiment Study on Asymmetric Constrained Meta Paths The experiments are done on the DBLP dataset. We evaluate the importance of authors and conferences simultaneously based on the meta path APC, which means authors publish papers on conferences. Two constrained meta paths ($APC|P.L =$ "DB" and $APC|P.L =$ "IR") are also included, which means authors publish DB(IR) field papers on conferences. Similarly, the experiments also include two baseline methods (i.e., PageRank and Degree) in above experiments with the same experimental process.

The top ten authors and conferences returned by these five methods are shown in Tables 4.2 and 4.3, respectively. As shown in Table 4.2, the ranking results of these methods on authors all are reasonable; however, the constrained meta paths can find the most influential authors in a certain field. For example, the top three authors of $APC|P.L =$ "DB" are Surajit Chaudhuri, Hector Garcia-Molina, and H.V. Jagadish, and all of them are very influential researchers in the database field. The

[2] http://arnetminer.org/.

Table 4.2 Top ten authors of different methods on DBLP dataset. The number in the parenthesis of the fifth column means the rank of authors in the whole ranking list returned by PageRank

Rank	*APC*	*APC*|*P.L* = "*DB*"	*APC*|*P.L* = "*IR*"	PageRank	Degree
1	Gerhard Weikum	Surajit Chaudhuri	W. Bruce Croft	W. Bruce Croft(23)	Philip S. Yu
2	Katsumi Tanaka	H. Garcia-Molina	Bert R. Boyce	Gerhard Weikum(24)	Gerhard Weikum
3	Philip S. Yu	H.V. Jagadish	Carol L. Barry	Philip S. Yu(25)	Divesh Srivastava
4	H. Garcia-Molina	Jeffrey F. Naughton	James Allan	Jiawei Han(26)	Jiawei Han
5	W. Bruce Croft	Michael Stonebraker	ChengXiang Zhai	H. Garcia-Molina(27)	H. Garcia-Molina
6	Jiawei Han	Divesh Srivastava	Mark Sanderson	Divesh Srivastava(28)	W. Bruce Croft
7	Divesh Srivastava	Gerhard Weikum	Maarten de Rijke	Surajit Chaudhuri(29)	Surajit Chaudhuri
8	Hans-Peter Kriegel	Jiawei Han	Katsumi Tanaka	H.V. Jagadish(30)	H.V. Jagadish
9	Divyakant Agrawal	Christos Faloutsos	Iadh Ounis	Jeffrey F. Naughton(31)	Jeffrey F. Naughton
10	Jeffrey Xu Yu	Philip S. Yu	Joemon M. Jose	Rakesh Agrawal(32)	Rakesh Agrawal

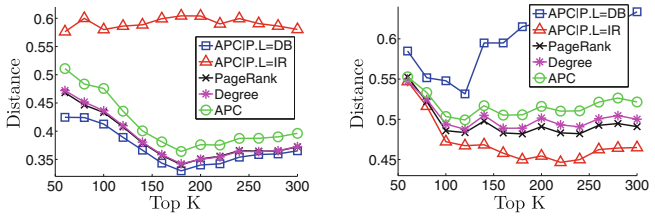
Table 4.3 Top ten conferences of different methods on DBLP dataset. The number in the parenthesis of the fifth column means the rank of conferences in the whole ranking list returned by PageRank

Rank	*APC*	*APC*|*P.L* = "*DB*"	*APC*|*P.L* = "*IR*"	PageRank	Degree
1	CIKM	ICDE	SIGIR	ICDE(3)	ICDE
2	ICDE	VLDB	WWW	SIGIR(4)	SIGIR
3	WWW	SIGMOD	CIKM	VLDB(5)	VLDB
4	VLDB	PODS	JASIST	CIKM(6)	SIGMOD
5	SIGMOD	DASFAA	WISE	SIGMOD(7)	CIKM
6	SIGIR	EDBT	ECIR	JASIST(8)	JASIST
7	DASFAA	ICDT	APWeb	WWW(9)	WWW
8	JASIST	MDM	WSDM	DASFAA(10)	PODS
9	WISE	WebDB	JCIS	PODS(11)	DASFAA
10	EDBT	SSTD	IJKM	JCIS(12)	EDBT

top three authors of $APC|P.L =$ "*IR*" are W. Bruce Croft, Bert R. Boyce, and Carol L. Barry, and they all have the high academic reputation in the information retrieval field. Similarly, as we can see in Table 4.3, HRank with constrained meta paths (i.e., $APC|P.L =$ "*DB*" and $APC|P.L =$ "*IR*") can clearly find the important conferences in DB and IR fields, while other methods mingle these conferences. For example, the most important conferences in the DB field are ICDE, VLDB, and SIGMOD, while the most important conferences in the IR field are SIGIR, WWW, and CIKM. Observing Tables 4.2 and 4.3, we can also find the mutual effect of authors and conferences. That is, an influential author published many papers in the important conferences, and vice versa. For example, W. Bruce Croft published many papers in SIGIR and CIKM, while Surajit Chaudhuri has many papers in SIGMOD, ICDE, and VLDB.

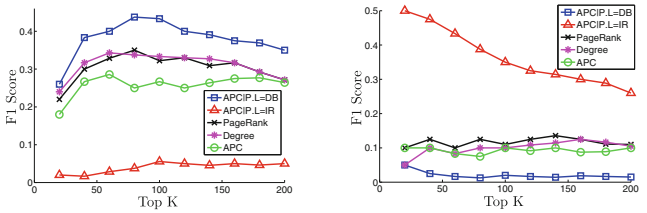
Quantitative Comparison Experiments To verify the effectiveness of these methods, we use the above *Distance* criterion to calculate the difference between their results and standard ranking lists crawled from Microsoft Academic Search. Figure 4.6 shows the differences of author ranking lists. We can observe the same phenomenon with above quantitative experiments again. That is, HRank with constrained meta paths achieve the best performances on their corresponding field. Meanwhile, they have the worst performances on other fields. In addition, compared to that of PageRank and Degree, the mediocre performances of HRank with meta path *APC* further demonstrate the importance of constrained meta path to capture the subtle semantics contained in heterogeneous networks. Similarly, we further evaluate the F1 accuracy of these methods according to the ground truth crawled from ArnetMiner. The results are shown in Fig. 4.7. Once again the results reveal the same findings that HRank can more accurately discover the authors ranking in a special field with the help of constrained meta path.

Experiments on Meta Path with Multiple Constraints Furthermore, we validate the effectiveness of meta path with multiple constraints. In the above experiments, we employ the constraint on the label of papers in HRank with the meta path *APC*. Here, we add one more constraint on conference. Specifically, by contrast to the constrained meta path $APC|P.L =$ "*DB*", we employ the paths $APC|P.L =$



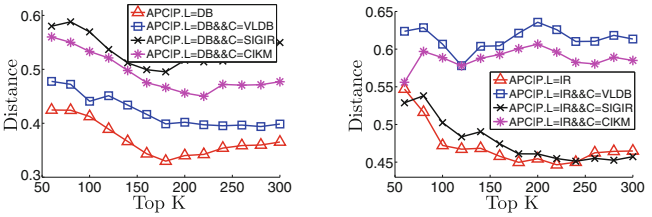
(a) DB field (b) IR field

Fig. 4.6 Distances between the candidate author ranking lists and the standard ranking lists on different fields on DBLP dataset. The ground truth is from Microsoft Academic Search



(a) DB field (b) IR field

Fig. 4.7 F1 accuracy of the ranking lists obtained by different methods on different fields on DBLP dataset. The ground truth is from ArnetMiner



(a) DB field (b) IR field

Fig. 4.8 Rank accuracy of HRank with different constrained meta paths on DBLP dataset

"*DB*"&&C = "*VLDB*", $APC|P.L$ = "*DB*"&&C = "*SIGIR*", and $APC|P.L$ = "*DB*"&&C = "*CIKM*", which mean authors publish DB field papers on specified conferences (e.g., VLDB, SIGIR, and CIKM). Similarly, we add the same conference constraints on the path $APC|P.L$ = "*IR*". Same with the above experiments, we calculate the rank accuracy of HRank with these constrained meta paths and the results are shown in Fig. 4.8.

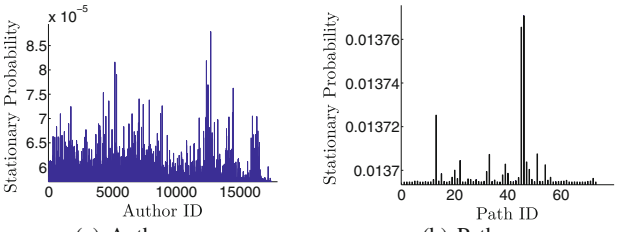
We know that HRank with the path $APC|P.L$ = "*DB*" ($APC|P.L$ = "*IR*") can reveal the influence of authors in the DB (IR) field. As ground truth, this ranking is based on the aggregation of many conferences related to the DB field. The added conference constraint in HRank further reveals the influence of authors in the specific conference of the field. So we can use the closeness to the ground truth to reveal the importance of a conference to that field. That is, if the ranking from a specific conference is quite closer to the ground truth rank, that can imply the conference is a dominating conference in that field. From Fig. 4.8a, we can find that the VLDB conference constraint (the blue curve) achieves the closest performances to the ground truth ranking, while the performances of the SIGIR conference constraint (the black curve) deviate most. So we can infer that the VLDB is more important than SIGIR in the DB field and the CIKM has the middle importance. Similarly, from Fig. 4.8b, we can infer that the SIGIR is more important than VLDB in the IR field. These findings comply with our common sense. As we know, although the VLDB and SIGIR both

are the top conferences in computer science, they are very important only in their research fields. For example, the VLDB is important in the DB field, while it is not so important in the IR field. The middle importance of the CIKM conference stems from the fact that it is a comprehensive conference including papers from both DB and IR fields. In addition, we can find that the SIGIR curve almost overlaps with the ground truth over the IR field, while the VLDB curve still has a gap with the ground truth over the DB field. We think the reason is that SIGIR is the main conference in the IR field, while in the DB field, there are also other important conferences, such as SIGMOD and ICDE. Overall, the experiments show that HRank with constrained meta path can not only effectively find the influential authors in each research field on a specified conference but also indirectly reveal the importance of conferences in the fields. It also implies that HRank can achieve accurate and subtle ranking results by flexibly setting the combination of constraints.

4.1.3.3 Co-ranking of Objects and Paths

Experiment Study on Co-ranking on Symmetric Constrained Meta Paths In this experiment, we will validate the effectiveness of HRank-CO to rank objects and symmetric constrained meta paths simultaneously. The experiment is done on ACM dataset. First, we construct a (2, 1)th order tensor X based on 73 constrained meta paths (i.e., $APA|P.L = L_j, j = 1 \cdots 73$). When the ith and the kth authors co-publish a paper together, of which the label is the jth label (i.e., ACM categories), we add one to the entries $x_{i,j,k}$ and $x_{k,j,i}$ of X. In this case, X is symmetric with respect to the index j. By considering all the publications, $x_{i,j,k}$ (or $x_{k,j,i}$) refers to the number of collaborations by the ith and the kth author under the jth paper label. In addition, we do not consider any self-collaboration, i.e., $x_{i,j,i} = 0$ for all $1 \leq i \leq 17{,}431$ and $1 \leq j \leq 73$. The size of X is $17{,}431 \times 73 \times 17{,}431$ where there are 91,520 nonzero entries in X. The percentage of nonzero entries is $4.126 \times 10^{-4}\%$. In this dataset, we will evaluate the importance of authors through the co-author relations, and meanwhile, we will analyze the importance of paths (i.e., which paths have the most contributions to the importance of authors).

Figure 4.9 shows the stationary probability distributions of authors and paths. It is obvious that some authors and paths have higher stationary probability, which implies



(a) Authors (b) Paths

Fig. 4.9 Stationary probability distributions of authors and constrained meta paths

Table 4.4 Top 10 authors and constrained meta paths (note that only the constraint (L_j) of the paths ($APA|P.L = L_j, j = 1 \ldots 73$) are shown in the third column of the table)

Rank	Authors	Constrained meta paths
1	Jiawei Han	H.3 (Information storage and retrieval)
2	Philip Yu	H.2 (Database management)
3	Christos Faloutsos	C.2 (Computer-communication networks)
4	Ravi Kumar	I.2 (Artificial intelligence)
5	Wei-Ying Ma	F.2 (Analysis of algorithms and problem complexity)
6	Zheng Chen	D.4 (Operating systems)
7	Hector Garcia-Molina	H.4 (Information systems applications)
8	Hans-Peter Kriegel	G.2 (Discrete mathematics)
9	Gerhard Weikum	I.5 (Pattern recognition)
10	D.R. Karger	H.5 (Information interfaces and presentation)

that these authors and paths are more important than others. Table 4.4 shows the top ten authors (left) and paths (right) based on their HRank values. We can find that the top ten authors all are influential researchers in the DM/IR fields, which conforms to our common senses. Similarly, the most important paths are related to DM/IR fields, such as $APA|P.L =$ "$H.3$" (information storage and retrieval) and $APA|P.L =$ "$H.2$" (database management). Although the conferences in ACM dataset are from multiple fields, such as DM/DB (e.g., KDD, SIGMOD) and computation theory (e.g., SODA, STOC), there are more papers from the DM/DB fields, which makes the authors and paths in the DM/DB fields ranked higher. We can also find that the influence of authors and paths can be promoted by each other. The reputation of Jiawei Han and Philip Yu come from their productive papers in the influential fields (e.g., H.3 and H.2). In order to observe this point more clearly, we show the number of co-authors of the top ten authors based on the top ten paths in Table 4.5. We can observe that there are more collaborations for top authors in the influential fields. For example, although Zheng Chen (rank 6) has more number of co-authors than Jiawei Han (rank 1), the collaborations of Jiawei Han focus on ranked higher fields (i.e., H.3 and H.2), so Jiawei Han has higher HRank score. Similarly, the top paths contain many collaborations of influential authors.

Experiment Study on Co-ranking on Asymmetric Constrained Meta Paths The experiments on the Movie dataset aim to show the effectiveness of HRank-CO to rank heterogeneous objects and asymmetric constrained meta paths simultaneously. In this case, we construct a third-order tensor X based on the constrained meta paths $AMD|M.T$. That is, the tensor represents the actor–director collaboration relations on different types of movies. When the ith actor and the kth director cooperate in a movie of the jth type, we add one to the entries $x_{i,j,k}$ of X. By considering all the cooperations, $x_{i,j,k}$ refers to the number of collaborations by the ith actor and the kth director under the jth type of movie. The size of X is $5324 \times 112 \times 551$, and there are 36,529 nonzero entries in X. The percentage of nonzero entries is $7.827 \times 10^{-4}\%$.

Table 4.5 Number that the top ten authors collaborate with others via the top ten constrained meta paths (note that only the constraints (L_j) of the paths ($APA|P.L = L_j, j = 1 \ldots 73$) are shown in the first row of the table)

Ranked author/CP	1 (H.3)	2 (H.2)	3 (C.2)	4 (I.2)	5 (F.2)	6 (D.4)	7 (H.4)	8 (G.2)	9 (I.5)	10 (H.5)
1 (Jiawei Han)	51	176	0	0	0	0	9	2	2	0
2 (Philip Yu)	51	94	0	0	9	0	3	0	13	0
3 (C. Faloutsos)	17	107	0	5	9	0	3	4	2	0
4 (Ravi Kumar)	73	27	0	3	13	0	18	5	0	0
5 (Wei-Ying Ma)	132	26	0	9	0	0	2	0	30	10
6 (Zheng Chen)	172	9	0	9	0	0	22	0	38	9
7 (H. Garcia-Molina)	23	65	3	0	0	0	1	0	0	4
8 (H. Kriegel)	19	28	5	0	0	0	6	0	7	4
9 (G. Weikum)	82	14	0	4	0	0	8	0	4	0
10 (D.R. Karger)	11	5	13	0	7	4	1	7	0	7

Table 4.6 Top 10 actors, directors, and meta paths on IMDB dataset (note that only the constraints (T_j) of the paths ($AMD|M.T = T_j, j = 1 \ldots 1591$) are shown in the fourth column)

Rank	Actor	Director	Conditional meta path
1	Eddie Murphy	Tim Burton	Comedy
2	Harrison Ford	Zack Snyder	Drama
3	Bruce Willis	Marc Forster	Thriller
4	Drew Barrymore	David Fincher	Action
5	Nicole Kidman	Michael Bay	Adventure
6	Nicolas Cage	Ridley Scott	Romance
7	Hugh Jackman	Richard Donner	Crime
8	Robert De Niro	Steven Spielberg	Sci-Fi
9	Brad Pitt	Robert Zemeckis	Animation
10	Christopher Walken	Stephen Sommers	Fantasy

Table 4.6 shows the top ten actors, directors, and constrained meta paths (i.e., movie type). We observe the mutual enhancements of the importance of objects and meta paths again. Basically, the results comply with our common senses. The top ten actors are well known, such as Eddie Murphy and Harrison Ford. Similarly, these directors are also famous in filmdom due to their works. These movie types obtained are the most popular movie subjects as well. In addition, we can observe the mutual effect of objects and paths one more time. As we know, Eddie Murphy and Drew Barrymore (rank 1, 4 in actors) are famous comedy and drama (rank 1, 2 in paths) actors. Harrison Ford and Bruce Willis (rank 2, 3 in actors) are popular thrill and action (rank 3, 4 in paths) actors. These higher ranked directors also prefer to those popular movie subjects. Furthermore, we also compare these results with the

recommended results from the IMDB website.[3] Although only a subset of movies in IMDB is included in our experiments, the 80% of the top 10 actors in our results are included in the set of the top 250 greatest movie actors in all time recommended by IMDB,[4] and the 50% of the top 10 directors in our results are included in the set of the top 50 favorite directors recommended by IMDB.[5] Moreover, most of movie types recommended by our method have high ranks in the popular types summarized by IMDB.[6] The more details of the HRank method and experimental results can be seen in [18].

4.2 Ranking-Based Clustering

4.2.1 Overview

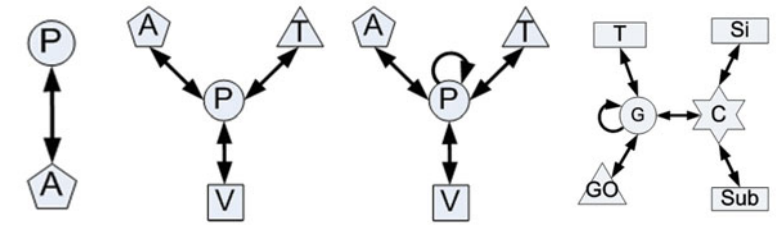
Recently, the link-based clustering attracts more and more attention, which usually groups objects that are densely interconnected but sparely connected with the rest of the network [11]. Also with the boom of search engine, object ranking [1, 5] becomes an important data mining task, which evaluates the importance of objects. Conventionally, clustering and ranking are two independent tasks and they are usually used separately. However, recent researches show that clustering and ranking can mutually promote each other and their combination makes more sense in many applications [21, 22]. If we know the important objects in a cluster, we can understand this cluster better; and the ranking in a cluster provides more subtle and meaningful information for clustering. Although it is a promising way to do clustering and ranking together, previous approaches are confined to a simple HIN with special structure. For example, Sun et al. validated the mutual improvement of clustering and ranking in bipartite network [21] (an example shown in Fig. 4.10a) and star-schema network [22] (an example shown in Fig. 4.10b). Shi et al. [27] integrated clustering and ranking in the hybrid network including heterogeneous and homogeneous relations. However, the data in real applications are usually more complex and irregular, which are beyond the widely used bipartite or star-schema network. For example, the bibliographic data (see an example in Fig. 4.10c) include not only heterogeneous relations but also homogeneous relations (e.g., self loop on P); the bioinformatics data [2] (see an example in Fig. 4.10d) have more complex structure, which includes multiple hub objects (e.g., C and G). So it is desirable to design effective ranking-based clustering algorithm for these complex and irregular HIN data. Broadly speaking, for HIN with arbitrary schema, we need to design a general solution to manage the objects and their relations, which is the basic for mining useful patterns on it.

[3] http://www.imdb.com/.

[4] http://www.imdb.com/list/ls050720698/.

[5] http://www.imdb.com/list/ls050131440/.

[6] http://www.imdb.com/list/ls050782187/?view=detail&sort=listorian:asc.



(a) bi-partite (b) star-schema (c) DBLP (d) SLAP

Fig. 4.10 Examples of heterogeneous information networks. The letters are the abbreviation of different types of objects (e.g., *P*: paper, *A*: author)

Obviously, it is more practical and useful to determine the underlying clusters and ranks on a general heterogeneous information network, but they are seldom exploited until now. When we integrate ranking and clustering on an HIN with arbitrary schema, it faces the following challenges. (1) A general HIN has more complex structure. For a simple HIN with a bipartite or star-schema structure, it is relatively easy to manage heterogeneous objects and build models. However, a general HIN may have arbitrary schema, beyond the bipartite or star-schema structure. Although an intuitive way is to decompose it into multiple simpler subnetworks, the issue is how we decompose the HIN without structural information loss and maintain the consistency among the decomposed subnetworks. (2) It is challenging to integrate the clustering and ranking in a complex heterogeneous network. We know that it is still a daunting task to separately do clustering and ranking on a general HIN. Therefore, it is more difficult to design an effective mechanism to combine these two tasks on the HIN.

In this chapter, we study the ranking-based clustering problem on a general HIN and propose a novel algorithm **HeProjI** to solve the **He**terogeneous network **Proj**ection and **I**ntegration of clustering and ranking tasks. In order to conveniently manage objects and relations in an HIN with arbitrary schema, we design a network projection method to project the HIN into a sequence of subnetworks without structural information loss, where the subnetwork may be a relatively simple bipartite or star-schema network. Moreover, an information transfer mechanism is developed to maintain the consistency across subnetworks. For each subnetwork, a path-based random walk method is proposed to generate the reachable probability of objects, which can be effectively used to estimate the cluster membership probability and the importance of objects. Through iteratively analyzing each subnetwork, HeProjI can obtain the steady and consistent clustering and ranking results. We perform a number of experiments on three real datasets to validate the effectiveness of HeProjI. The results show that HeProjI not only achieves better clustering and ranking accuracy compared to well-established algorithms, but also effectively handles complex HIN which cannot be handled by previous methods.

4.2.2 Problem Formulation

In this section, we give the problem definition and some important concepts used in this chapter.

Definition 4.7 (*General heterogeneous information network*) Given a schema $\mathtt{A} = (\mathtt{T}, \mathtt{R})$ which consists of a set of entities type $\mathtt{T} = \{\mathrm{T}\}$ and a set of relations $\mathtt{R} = \{\mathrm{R}\}$, a general information network is defined as a graph $\mathtt{G} = (\mathrm{X},\mathrm{E})$ with an object type mapping function $\tau : X \to \mathtt{T}$ and link type mapping function $\psi : E \to \mathtt{R}$. Each object $|\mathtt{T}| > 1$ or the types of relations $|\mathtt{R}| > 1$, the network is called **heterogeneous information network**; otherwise, it is a **homogeneous information network**.

Figure 4.10 shows the schema of several HIN examples. The bipartite network in Fig. 4.10a only includes two types of objects, and the widely used star-schema network [16, 22, 24] in Fig. 4.10b organizes objects in HIN with one target type and several attribute types. However, a general heterogeneous information network may be more complex and irregular. It may not only include homogeneous or heterogeneous relations, but also include multiple hub objects. Figure 4.10d shows such a general HIN example. The object G has heterogeneous relations (e.g., $G \to GO$ and $G \to C$) as well as homogeneous relations (e.g., $G \to G$). Moreover, the network is beyond the star-schema because of multiple hub objects (e.g., G and C). It is clear that bipartite graph and star-schema network are the special case of a general HIN.

For a general HIN, it is difficult to manage objects and relations in the network. Although we can project it into several homogeneous networks through assigning meta paths as reference [3] did, it will loss much information among different-typed objects. We know that, as the special case of HIN, the bipartite and star-schema networks are relatively easy to manage objects and relations in the network. So a basic idea of handling a general HIN is to decompose it into simpler networks. Following this idea, we design a novel HIN projection method. Specifically, we can select one type (called pivotal type) and its connected other types (called supportive type). These types and their relations constitute the schema of a projected subnetwork of original HIN. Formally, it can be defined as follows:

Definition 4.8 (*Projected subnetwork*) For an HIN with schema $\mathtt{A} = (\mathtt{T}, \mathtt{R})$, its projected subnetwork has the schema $\mathtt{A}' = (\mathtt{T}', \mathtt{R}')$ where $\mathtt{T}' \subset \mathtt{T}$, $\mathtt{R}' \subset \mathtt{R}$, $\mathtt{T}'$ includes one **pivotal type** (denoted as P) and other types connected with P (called **supportive type**, denoted as $\mathtt{S} = \{S\}$). $\mathtt{R}'$ includes the heterogeneous relations between P and S and homogeneous relations among P (if existing).

A projected subnetwork can be denoted as $P - \mathtt{S}$. The $X^{(P)}$ is the object set of pivotal type, and $X^{(S)}$ represents the object set of supportive type S. For convenience, the projected subnetwork is also called subnetwork which can be represented with its pivotal type P. For example, Fig. 4.11c shows the projected subnetwork $G - \{C, T, GO\}$ with type G object (the one in red) as the pivotal type, while types C, T and GO are the supportive types as they are object types connected to object type G. Similarly, Fig. 4.11b and d shows the projected subnetworks with pivotal type objects GO and C, respectively.

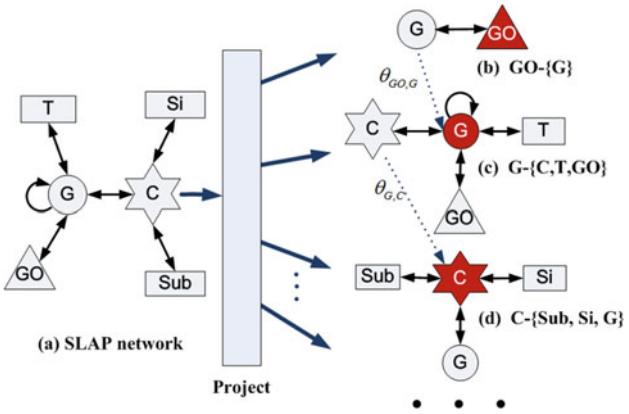


Fig. 4.11 An example of HIN projection. The pivotal type is marked with *red color*. The *dot line* represents the information transfer among subnetworks

It is clear that an HIN can be projected into a sequence of subnetworks through selecting different pivotal types. So we define the HIN projection concept as follows.

Definition 4.9 (*HIN projection*) An HIN with t types of objects can be projected into an ordered set of t projected subnetworks by successively selecting one of the t types as pivotal type.

Figure 4.11 shows a projection example of SLAP network, a bioinformatics dataset [17]. Through successively selecting the six object types (GO, G, C and so on) as pivotal type, the SLAP network is projected into a sequence of six subnetworks. It is clear that the HIN projection has the following properties.

Property 4.1 *HIN projection is a structure–information-lossless network decomposition.*

According to Definition 4.9, all objects and relations in original HIN are in the projected subnetworks. That is to say, the HIN can be reconstructed from the set of projected subnetworks.

Property 4.2 *Each projected subnetwork in HIN projection should be a bipartite graph or a star-schema network (with self loop).*

According to Definition 4.8, if there are two types of objects in the subnetwork, it is a bipartite graph; otherwise, it is a star-schema network. Note that, different from the conventional bipartite and star-schema networks, the pivotal type in subnetworks may include the homogenous relation (i.e., self loop).

Property 4.3 *HIN projection is not unique for a general HIN.*

An HIN has different projection sequences through selecting different orders of pivotal types. For example, the SLAP network in Fig. 4.11 has the projection sequences: $GO-G-C-Si-Sub-T$, $T-G-GO-C-Si-Sub$, and so on. In fact, an HIN with t types of objects has the $t!$ projection sequences in all.

Assume that J represents a type in type set $\{T\}$. The object set can be denoted as $X=\{X^{(J)}\}$, and $X^{(J)}=\{X_p^{(J)}\}$ where $X_p^{(J)}$ is the object $p \in X^{(J)}$ (i.e., $\tau(p)=J$). The relations among objects include two types (homogeneous and heterogeneous relations), which can be represented by the two types of matrices, **homogeneous** and **heterogeneous relation matrices**, respectively. If type J has homogeneous relation (e.g., the self loop on P in Fig. 4.10c), the homogenous relation matrices can be written as $H^{(J)}$, where $H_{pq}^{(J)}$ denotes the relation between $X_p^{(J)}$ and $X_q^{(J)}$. If two types (I and J) have heterogeneous relation (e.g., $P-A$ in Fig. 4.10c), the heterogeneous relation matrices can be written as $H^{(I,J)}$, where $H_{pq}^{(I,J)}$ denotes the relation between $X_p^{(I)}$ and $X_q^{(J)}$. Correspondingly, we have **homogeneous transition matrix** $M^{(J)}$ and **heterogeneous transition matrix** $M^{(I,J)}$. It is clear that the transition matrix $M^{(I,J)}$ can be derived from the relation matrix $H^{(I,J)}$ by $M^{(I,J)}=D^{(I,J)^{-1}}H^{(I,J)}$, where $D^{(I,J)}$ is the diagonal matrix with the diagonal value equaling to the corresponding row sum of $H^{(I,J)}$. Similarly, $M^{(J)}=D^{(J)^{-1}}H^{(J)}$. Taking Fig. 4.10c as example, $M^{(P)}$ is the transition probability matrix of the citation relation $H^{(P)}$, and $M^{(A,P)}$ is the transition probability matrix of the $A-P$ relation $H^{(A,P)}$. For given network structure, we can derive the homogeneous and heterogeneous transition matrices. In the following section, we consider that the transition matrices are known.

Different from conventional clustering in homogeneous networks, cluster in HIN should include different types of objects, where these objects share the same semantic meaning. For example, in bibliographic data, a cluster about data mining area includes venues, authors, and papers in this field. For each type of objects $X^{(J)}$, we define the **membership matrix** $B^{(J|C_k)} \in [0,1]^{|X^{(J)}|\times|X^{(J)}|}$, which is a diagonal matrix whose diagonal value represents the membership probability of $X_p^{(J)}$ belonging to the cluster C_k. Note that the sum of membership probability of $X_p^{(J)}$ in K clusters is 1 (i.e., $\sum_{k=1}^{K} B_{pp}^{(J|C_k)}=1$). Now, we can formulate the problem of clustering on a general HIN as follows: Given a heterogeneous network $G=(X,E)$ and the semantic cluster number K, our goal is to find a clusters set $\{C_k\}_{k=1}^{K}$, where C_k is defined as $C_k = \{\{B^{(J|C_k)}\}_{J\in\{T\}}\}$. In this way, it is a soft clustering. That is, an object p in $X^{(J)}$ can belong to several clusters, and it is in a cluster C_k with the probability $B_{pp}^{(J|C_k)}$. Moreover, a cluster C_k can contain all kinds of objects.

4.2.3 The HeProjI Algorithm

Through the HIN projection, it will become much easier to analyze the HIN through handling a set of simple projected subnetworks, since these subnetworks are bipartite or star-schema networks. However, it may result in a troublesome business: how to maintain the consistency among different subnetworks. To solve it, we design

an information transfer mechanism which inherits a portion of information from other subnetworks to current one. In order to integrate the clustering and ranking in a uniform framework, a model is required to flexibly support these two tasks. Following this idea, we build a probabilistic model to estimate the probability of supportive and pivotal objects in each subnetwork. Moreover, the probability of objects can effectively infer the clustering information and represent the importance of objects.

4.2.3.1 Framework of HeProjI Algorithm

Specifically, we first project the original HIN into a sequence of subnetworks and then randomly assign the pivotal objects of the first subnetwork into K clusters (i.e., initialize $\{C_k\}_{k=1}^K$). For each subnetwork, a path-based random walk method is proposed to estimate the reachable probability of supportive objects in each cluster C_k, and then, a generative model is used to obtain the probability of pivotal objects. After that, an EM algorithm is employed to estimate the posterior probability of objects (i.e., the clustering information $\{C_k\}_{k=1}^K$). According to probability of objects, we can also calculate their ranking in each cluster. The above step is repeated until convergence. In the iterative process, the clustering and ranking can mutually promote each other until they reach a steady result. The basic framework of HeProjI is shown in Algorithm 4.2. In the following sections, we will present these operations in detail.

4.2.3.2 Reachable Probability Estimation of Objects

Basic idea As we have noted that the built probabilistic model can not only support the clustering and ranking tasks but also maintain the consistency among subnetworks, so the design of the model should obey the following two rules: (1) PageRank principle. In order to support the ranking task, the probability of objects should be able to

Algorithm 4.2 HeProjI: Detecting K clusters on HIN

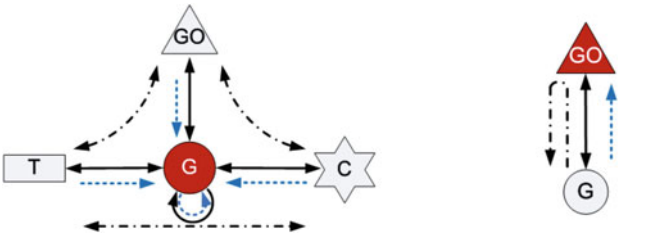
Input:
Cluster number K and transition probability matrix M.
Output:
Membership probability $B^{(J|C_k)}$ of objects on each cluster$\{C_k\}_{k=1}^K$

Project the HIN into a sequence of sub-networks
Randomly initialize the membership probability $B^{(J|C_k)}$
repeat
  Select the projected sub-network $(P - S)$ in order
  for cluster $C_k \in C$ **do**
    Establish the probability of supportive objects: $Pr(X^{(S)}|C_k)$
    Generate the probability of pivotal objects: $P(X^{(P)}|C_k)$
    Estimate the posterior probability of objects: $P(C_k|X^{(P)})$, $P(C_k|X^{(S)})$
  end for
  Rank the objects: $Rank(X^{(P)}|C_k)$, $Rank(X^{(S)}|C_k)$
until the membership probability obtains convergence

reflect their ranks. In other words, the probability of objects should be positively correlated with the node degree. (2) Consistency principle. In order to maintain the consistency among subnetworks, an effective mechanism should be designed to transfer appropriate information among subnetworks.

For the first rule (i.e., PageRank principle), the random walk is an apparent solution. However, it is traditionally used in homogeneous networks [1, 5]. Although it is also used in bipartite graph [28], it is seldom applied in HIN. Sun et al. [22] employed it to estimate the probability of attribute objects in a star-schema network, while it is confined to two types of objects. Heterogeneous objects and link semantics make it difficult to directly employ random walk in HIN. In a projected subnetwork, there are different types of supportive objects and they are connected through pivotal objects. So the random walk among objects should follow the specified paths. That is, the random walkers among supportive objects would need to pass through the pivotal objects. As a consequence, we need to estimate the probability of supportive and pivotal objects separately. The reachable probability of a supportive object can be calculated as the sum of the probability of walkers from other supportive objects walking to it through the pivotal type. The probability of pivotal objects can be generated through its reachable supportive objects. Because the bipartite network only contains one supportive type, the probability of supportive object can be calculated by the sum of probability of walkers from the same type of objects walking to it through the pivotal type. Figure 4.12 shows the probability estimation process. The reachable probability of type C can be calculated by random walkers wandering from type GO and T to type C through type G in Fig. 4.12a.

For the second rule (consistency principle), it is an intuitive idea to transfer information among subnetworks. However, what and how do we transfer? It is clear that the subnetworks are overlapped. If we transfer the information of any overlapping types, the model may be hard to control, since two subnetworks may have many overlapping types and one type may appear in many subnetworks. If we do clustering on each subnetwork individually, it is difficult to map clusters among subnetworks. We know that the random walk among all supportive objects passes through the pivotal objects. So we only need to transfer the information of pivotal type, and then,



(a) star-schema network (b) bipartite network

Fig. 4.12 Illustration of the probability estimation process for supportive and pivotal objects. The *black dash-dot line* represents the random walk process among supportive objects, and the *blue dotted line* represents the generative process of pivotal objects

the information can be propagated to other supportive objects by random walkers. In order to maintain the clustering consistency during the iteration, we let the pivotal objects in the current subnetwork inherit a portion of clustering information from previous subnetworks with a controlling parameter. The dot line in Fig. 4.11 shows two information inheritance examples. Specifically, the information on object G calculated in Fig. 4.11b is passed on to the calculation of the pivotal object G in Fig. 4.11c which affects the calculation of object C, while the information on object C is then passed on to the calculation of pivotal object C in Fig. 4.11d.

Reachable Probability for Supportive Objects First, we estimate the probability of supportive objects. The path-based random walk process is formulated with matrix representation. We use $M^{(S^I,S^J|P,C)}$ to represent the probability transition matrix from supportive type S^I to type S^J passing pivotal type P in the subnetwork C. $M^{(S^I,S^J|P,C)}$ can be calculated as follows:

$$M^{(S^I,S^J|P,C)} = M^{(S^I,P|C)} \times M^{(P,S^J|C)} \tag{4.14}$$

where $M^{(S^I,P|C)}$ is the transition matrix from S^I to P (i.e., $M^{(S^I,P)}$). Compared to conditional transition matrix $M^{(S^I,S^J|P,C_k)}$ defined below, $M^{(S^I,S^J|P,C)}$ is also called the global transition matrix, which is fixed for the subnetwork C. For example, in Fig. 4.12a, the global transition matrix $M^{(T,GO|G,C)}$ means the transition probability from type T to GO through G on the subnetwork $G - \{T, C, GO\}$. In the proposed model, the global probability of objects is important information to smooth the probability of pivotal objects (see Eq. 4.21 for more details).

When considering the clustering information, the transition matrices among supportive objects should be adjusted according to clusters. The clustering information can be represented by the membership matrix of pivotal objects, so the conditional transition matrix from S^I to S^J through P in the cluster C_k (i.e., $M^{(S^I,S^J|P,C_k)}$) can be defined as follows:

$$M^{(S^I,S^J|P,C_k)} = M^{(S^I,P|C)} \times B^{(P|C_k)} \times M^{(P,S^J|C)} \tag{4.15}$$

where $B^{(P|C_k)}$ is the membership of pivotal objects on cluster C_k.

The above transition matrices only consider the clustering information in the current subnetwork, which may cause the inconsistency among different subnetworks. For example, in the bibliographic data shown in Fig. 4.10c, clustering on the subnetwork $P - \{A, V, T\}$ may focus on research areas, while clustering on the subnetwork $A - \{P\}$ may more concern about co-author relations. In order to keep the clustering consistency among subnetworks, we can inherit a portion of cluster information from previous subnetworks. Only the clustering information of pivotal type is inherited from previous networks, and it is integrated with current clustering information of pivotal type. The reason why the simple mechanism work is that the pivotal objects, as hub node, can propagate the clustering information to all supportive objects. The transition matrices can be redefined as:

$$B''^{(P|C_k)} = \theta_{S,P} \times B'^{(P|C_k)} + (1 - \theta_{S,P}) \times B^{(P|C_k)} \tag{4.16}$$

$$M^{(S^I,S^J|P,C_k)} = M^{(S^I,P|C)} \times B''^{(P|C_k)} \times M^{(P,S^J|C)} \tag{4.17}$$

where $B'^{(P|C_k)}$ is the inherited membership matrix when the type P serves as a supportive type in the subnetwork whose pivotal type is S, and the $\theta_{S,P}$ is a learning rate parameter that controls the ratio of information inheritance from previous subnetwork (pivotal type is S) to current one (pivotal type is P). The dot line in Fig. 4.11 illustrates the two examples of information inheritance. The new transition matrix has the following advantages: (1) It transfers the clustering information among subnetworks, which keeps the consistency of subnetworks, and (2) it helps to speed up the convergence, since the priori clustering information is adopted. For a bipartite network, the transition probability matrix can be denoted as $M^{(S^I,S^I|P,C_k)}$, which has the same calculation mechanism.

The conditional probability of supportive type S^J on subnetwork C and cluster C_k is denoted as $Pr(X^{(S^J)}|C) \in [0, 1]^{1\times|X^{(S^J)}|}$ and $Pr(X^{(S^J)}|C_k) \in [0, 1]^{1\times|X^{(S^J)}|}$. Inspired by the PageRank [1], the probability of one type of objects is decided by the reachable probability from other types of objects through pivotal objects. So the conditional probability of supportive type S^J can be defined as follows.

$$Pr(X^{(S^J)}|C) = \sum_{S^I\in S, S^I\neq S^J} Pr(X^{(S^I)}|C) \times M^{(S^I,S^J|P,C)} \tag{4.18}$$

$$Pr(X^{(S^J)}|C_k) = \sum_{S^I\in S, S^I\neq S^J} Pr(X^{(S^I)}|C_k) \times M^{(S^I,S^J|P,C_k)} \tag{4.19}$$

The calculation is an iterative process, and $Pr(X^{(S^J)}|C_k)$ is initialized as the even value at the first iteration. For a bipartite network, random walkers start from type S^J and end up with the same type through the pivotal type P. The probability of supportive type S^J, $Pr(X^{(S^J)}|C_k)$ can be defined as $Pr(X^{(S^J)}|C_k) = Pr(X^{(S^J)}|C_k) \times M^{(S^J,S^J|C_k)}$.

Reachable Probability for Pivotal Objects Then, we estimate the probability of pivotal objects. We can consider that the pivotal objects are generated by adjacent supportive objects, so a generative model can be adopted here. The probability of pivotal objects comes from two parts: heterogeneous and homogeneous relations (if the pivotal type has self loop). For heterogeneous relations, the heterogeneous probability of pivotal object p in the subnetwork C (i.e., $Pr(X_p^{(P)}|C)$) can be calculated as follows:

$$Pr(X_p^{(P)}|C) = \prod_{S^J\in S} \prod_{q\in N(p)} Pr(X_q^{(S^J)}|C) \tag{4.20}$$

where $N(p)$ is the set of neighbors of object p in the subnetwork. It means that the pivotal object p is generated by the different types of adjacent supportive objects.

Then, we consider the probability of pivotal object p in a cluster C_k (i.e., $Pr(X_p^{(P)}|C_k)$). Similarly, the probability is also generated from the adjacent supportive objects in the cluster C_k. In addition, we add the global probability of pivotal object $X_p^{(P)}$ to smooth the probability:

$$Pr(X_p^{(P)}|C_k) = \lambda \prod_{S^J \in Sq \in N(p)} \prod Pr(X_q^{(S^J)}|C_k) + (1-\lambda) Pr(X_p^{(P)}|C) \tag{4.21}$$

where the smooth parameter λ represents the portion of global probability. The smooth operation is an important component due to following reasons: (1) It prevents pivotal objects from accumulating into minority clusters, which helps to improve the clustering accuracy, and (2) it makes the probability change in pivotal objects more steady, which can improve the stability of HeProjI. The experiments in Sect. 5.7 also validate the importance of smooth operation.

For homogeneous relations (i.e., the pivotal object has self loop), we can calculate the cluster-based homogeneous transition probability for pivotal type as follows:

$$M^{(P|C_k)} = M^{(P|C)} \times B^{(P|C_k)} \tag{4.22}$$

$M_{.p}^{(P|C_k)}$ denotes the sum of transition probability of other pivotal objects reaching p in cluster C_k, which represents the importance of object p to some extent.

When considering the homogeneous relations (if existing), the probability of pivotal object p is generated by the heterogeneous and homogeneous relations, so it can be calculated as follows:

$$P(X_p^{(P)}|C_k) = Pr(X_p^{(P)}|C_k) \times M_{.p}^{(P|C_k)}. \tag{4.23}$$

4.2.3.3 Posterior Probability for Objects

In order to determine the membership of objects, we need to estimate posterior probability of objects. In each subnetwork, there are two kinds of objects (i.e., pivotal and supportive objects). Because pivotal objects are the hub of subnetwork that integrate supportive objects and contain complete semantic information, we first estimate the posterior probability of pivotal objects, and then, the posterior probability of supportive objects is decided by that of pivotal objects.

Now, we consider how to estimate the posterior probability of pivotal objects $P(C_k|X^{(P)})$. According to the Bayesian rule, $P(C_k|X^{(P)}) \propto P(X^{(P)}|C_k) \times P(C_k)$. Since the cluster size $P(C_k)$ is unknown, we need to estimate an appropriate $P(C_k)$ to balance the cluster size. We use the $P(C_k)$ that maximizes the likelihood of generating pivotal objects in different clusters. The likelihood of pivotal objects is defined as:

$$logL = \sum_{p \in X^{(P)}} log[\sum_{k=1}^{K} P(X_p^{(P)}|C_k) \times P(C_k)]. \tag{4.24}$$

An EM algorithm can be utilized for the latent $P(C_k)$ by maximizing the $logL$. We can derive the Eqs. 4.25 and 4.26. Initially, we set the $P(C_k)$ with even values and then repeat the E step (i.e., Eq. 4.25) and M step (i.e., Eq. 4.26) to iteratively update the latent cluster probability until the $P(C_k)$ obtains convergence.

$$P^t(C_k|X^{(P)}) \propto P(X^{(P)}|C_k) \times P(C_k) \tag{4.25}$$

$$P^{t+1}(C_k) = \sum_{p \in X^{(P)}} P^t(C_k|X_p^{(P)}) \times \frac{1}{|X^{(P)}|} \tag{4.26}$$

Next, we estimate the posterior of supportive objects. The basic idea is that the posterior probability of supportive objects comes from its pivotal neighborhoods. We define it as follows:

$$P(C_k|X_q^{(S^J)}) = \sum_{p \in N(q)} P(C_k|X_p^{(P)}) \times \frac{1}{|N(q)|} \tag{4.27}$$

where $P(C_k|X_q^{(S^J)})$ is the probabilities of supportive object $X_q^{(S^J)}$ belonging to cluster C_k; $N(q)$ is the neighbor set of supportive object q. It means that the posterior probability of supportive object $X_q^{(S^J)}$ is the average value of its pivotal neighborhoods.

4.2.3.4 Ranking for Objects

Since the probability model obeys the PageRank principle, we can regard the conditional probability of objects as their ranks.

$$Rank(X^{(J)}) \approx P(X^{(J)}|C_k) \tag{4.28}$$

Because the conditional probability $P(X^{(J)}|C_k)$ in HeProjI is estimated by the random walk process, it may prefer to assign a higher probability to an object with a higher degree. However, in some applications, the link number-based measure is not proper. For example, advertisement webpage may have many poor value links (i.e., high degree but low rank).

If we know the additional information of objects, which can be used to measure the importance of objects, we can integrate the information into the proposed method and then get the more reasonable rank. Based on the conditional probability of objects, we propose a general ranking method for objects as follows:

$$Rank(X^{(J)}) = AI(X^{(J)}) \times P(X^{(J)}|C_k) \tag{4.29}$$

where the $AI(X^{(J)})$ is the additional importance measure (AI) of objects $X^{(J)}$. For example, in bibliographic network, the importance of a paper is decided by its citations to a large extent, and the AI can be a measure that is proportion to citations. We

can also propagate the AI information to adjacent objects by transition probability matrix. It is denoted as follows:

$$Rank(X^{(I)}|C_k) = Rank(X^{(J)}|C_k) \times M^{(J,I)}. \tag{4.30}$$

4.2.4 Experiments

In this section, we evaluate the effectiveness of HeProjI and compare it with several state-of-art methods on three real datasets. In experiments, we use two real information networks: DBLP and SLAP. The schemas of these two networks are shown in Fig. 4.10c and d. In addition, we extract two different-scaled subsets of the DBLP which are called DBLP-S and DBLP-L, respectively. The DBLP-S is a small-size dataset which includes three research areas: database (DB), data mining (DM), and information retrieval (IR). While the DBLP-L is a large dataset which includes eight areas.

4.2.4.1 Clustering Effectiveness Study

In this section, we study the clustering effectiveness of HeProjI through comparing it with other well-established algorithms.

The first experiment is done on DBLP dataset, since this dataset has a relatively simple structure and is suitable for comparison with previous algorithms. The representative algorithms are included in experiments, which are summarized as follows:

- HeProjI. It is the proposed algorithm.
- HeProjI$_{\backslash S}$. It is HeProjI without considering the smooth information from general network (i.e., λ is 1 in Eq. 4.21).
- HeProjI$_{\backslash I}$. It is HeProjI without considering inheriting information from other subnetworks (i.e., Θ is 0 in Eq. 4.16).
- ComClus [27]. It is a ranking-based clustering method designed for the star-schema network with self loop.
- NetClus [22]. It is a ranking-based clustering method designed for the star-schema network without self loop.
- iTopicModel [20]. It integrates topic model and heterogeneous link information, so it can be used to do clustering in HIN.
- NetPLSA [9]. It regularizes a statistical topic model with a harmonic regularizer based on a graph structure.

The clustering quality is measured by the fraction of vertices identified correctly, FVIC [11, 15], which evaluates the average matching degree by comparing each predicting cluster with the most matching real cluster. The larger the FVIC is, the better the partition is. HeProjI, ComClus, and NetClus can be applied to DBLP dataset directly. For NetClus, we do not consider the self loop of type P, since

Table 4.7 Clustering accuracy for DBLP dataset

Accuracy		Paper (DBLP-S)	Venue (DBLP-S)	Author (DBLP-S)	Paper (DBLP-L)
HeProjI	Mean/Dev.	**0.857**/0.043	**0.823**/0.047	**0.725**/0.034	**0.603**/0.071
HeProjI$_{\backslash S}$	Mean/Dev.	0.781/0.077	0.753/0.069	0.698/0.057	0.566/0.113
HeProjI$_{\backslash I}$	Mean/Dev.	0.703/0.053	0.681/0.045	0.605/0.039	0.507/0.083
ComClus	Mean/Dev.	0.764/0.020	0.775/0.027	0.690/0.015	0.576/0.024
NetClus	Mean/Dev.	0.742/0.063	0.718/0.065	0.689/0.051	0.566/0.104
iTopicModel	Mean/Dev.	0.512/0.072	0.762/0.094	0.587/0.073	0.361/0.167
NetPLSA	Mean/Dev.	0.466/0.047	0.565/0.081	0.316/0.023	0.338/0.092

NetClus cannot solve it. Note that RankClus [21] is not included here, because it only solves the bipartite network. Moreover, for iTopicModel and NetPLSA, we make a homogeneity assumption of links so that it can be applied to this dataset. The smoothing parameter λ in HeProjI is fixed at 0.9. All learning rate Θ are fixed at 0.3. In HeProjI, the projection sequence is $P - A - C - T$. The parameters in other algorithms are set with the suggested values in their literals.

From the results shown in Table 4.7, we can observe that HeProjI achieves the best accuracy and lower standard deviation on all objects. HeProjI$_{\backslash S}$ also has good performances. However, due to omitting the smoothing operation, it has worse performances and stability when compared to HeProjI. The performances of HeProjI$_{\backslash I}$ degrade greatly, since it does not inherit clustering information from other subnetworks. In this condition, HeProjI$_{\backslash I}$ analyzes these subnetworks independently, so the inconsistency among subnetworks causes its bad performances. NetClus and ComClus both have respectable results. However, the absence of citation information among papers may lead to NetClus's worse performances when it is compared with ComClus. The iTopicModel and NetPLSA methods ignore the heterogeneity of objects and relations, so their performances are bad.

For SLAP network, contemporary methods cannot solve it directly. In order to compare with other algorithms, we convert the SLAP network into a homogeneous network through ignoring the heterogeneity of objects. As a comparison algorithm, the classical spectral clustering algorithm, NCut [19], is run on the homogeneous network. The projection sequence is $GO - G - C - T - Sub - Si$. HeProjI uses the same parameters with the above experiments, except the learning rate $\Theta[\theta_{G,GO}, \theta_{GO,G}, \theta_{G,C}, \theta_{G,T}, \theta_{C,Sub}, \theta_{C,Si}] = [0.3, 0.5, 0.7, 0.7, 0.7, 0.7]$. The results are shown in Table 4.8. It is clear that HeProjI performs much better than NCut. We know that there are distinct differences on different types of objects and relations, e.g., 70,672 links in $G - C$ relation and 2222 links in $G - GO$ relation. If we do not consider object types, as NCut does, the clusters may be seriously unbalanced, which results in the bad performances of NCut.

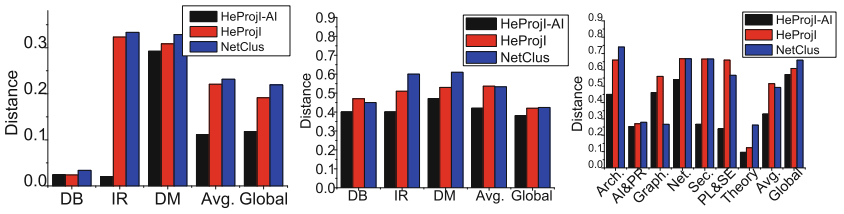
Table 4.8 Clustering accuracy for SLAP dataset

Accuracy	HeProjI		NCut	
	Mean	Dev.	Mean	Dev.
Gene	**0.68**	0.057	0.355	0.165
Chemical compound	**0.437**	0.031	0.307	0.091
Gene ontology	**0.557**	0.026	0.261	0.088
Tissue	**0.407**	0.066	0.293	0.09
Side effect	**0.548**	0.098	0.25	0.056
Substructure	**0.481**	0.053	0.314	0.102

4.2.4.2 Ranking Effectiveness Study

To evaluate the ranking effectiveness of HeProjI, we make a ranking accuracy comparison between HeProjI and NetClus. We utilize the venues rank recommended by Microsoft Academic Search [10] as the ground truth. In order to measure the quality of the ranking result, we employ the *Distance* criterion proposed in [12], which computes the differences between two ranking lists of the same set of objects. The criterion not only measures the number of mismatches between two lists but also gives a big penalty term to top mismatch objects in the lists. The smaller *Distance* means the better performance.

Three algorithms are tested on the DBLP dataset. In addition to NetClus, there are two versions of HeProjI (HeProjI with/without AI). The citations of paper are used as the AI measure. We extract the top 5 and 10 venues in different research areas and then calculate the *Distance* measure for them. Additionally, we also compare the accuracy of the global rank on both HeProjI and NetClus. The comparison results are shown in Fig. 4.13. We can find that two versions of HeProjI achieve better rank performances compared with NetClus in the most cases, since their *Distance* get lower values. Moreover, the HeProjI-AI performs better than HeProjI. In DBLP



(a) Top 5 on DBLP-S (b) Top 10 on AI&PR of DBLP-L (c) Top 10 on DBLP-L

Fig. 4.13 Ranking accuracy comparison on top venues (the smaller *Distance*, the better performance)

dataset, the citation information of papers (i.e., AI) reflects the quality of the papers to a large extent. So integrating the AI in HeProjI helps to improve the rank accuracy of papers. Moreover, the citation information can also promote the ranking accuracy of venues through the $P - V$ relation (see Eq. 4.30). So HeProjI-AI achieves the best ranking performances.

4.2.4.3 Case Study

We compare the ranking effectiveness of HeProjI and NetClus with a case study on DBLP dataset. We use the global rank to prove the ranking effectiveness of the HeProjI method. Table 4.9 shows the top 15 venues ranked by HeProjI and NetClus on DBLP-S. From these results, the ranks of venues generated by HeProjI-AI more conform to the intuition. Although it is hard to rank conferences across different areas, the order within each area is more or less established, and the HeProjI-AI confirms with that order. For example, in the DB area, it is SIGMOD, VLDB, and ICDE, while in the data mining area, it is KDD, ICDM, and PKDD. However, there are some out of order venues generated by NetClus. For example, among the database conferences, SIGMOD is ranked after VLDB and ICDE. Because NetClus cannot combine additional AI information (i.e., the citations of papers) and tends to get the rank which is proportion to its link number, it has the tendency to rank a good venue publishing a smaller number of papers with a lower rank (e.g., PODS) and a venue publishing a larger number of papers with higher rank (e.g., DEXA). Besides, for HeProjI which does not consider AI information, the rank of venues is basically proportional to their links, since the probability of objects is generated by a random walk-based method. The experiments reflect that the HeProjI method can flexibly

Table 4.9 Top 15 venues in 3 clusters on DBLP-S

Rank		1	2	3	4	5	6	7	8
HeProjI - AI	Venue	SIGMOD	VLDB	SIGIR	ICDE	KDD	PODS	WWW	CIKM
	#Papers	2428	2444	2509	2832	1531	940	1501	2204
HeProjI	Venue	ICDE	SIGIR	VLDB	SIGMOD	CIKM	DEXA	KDD	WWW
	#Papers	2832	2509	2444	2428	2204	1731	1531	1501
NetClus	Venue	VLDB	ICDE	SIGMOD	SIGIR	KDD	WWW	CIKM	ICDM
	#Papers	2444	2832	2428	2509	1531	1510	2204	1436
Rank		9	10	11	12	13	14	15	...
HeProjI-AI	Venue	ICDM	EDBT	PKDD	WSDM	PAKDD	DEXA	WebDB	
	#Papers	1436	747	680	198	1030	1731	972	...
HeProjI	Venue	ICDM	PAKDD	PODS	EDBT	PKDD	ECIR	WSDM	
	#Papers	1436	1030	1436	747	680	575	198	...
NetClus	Venue	PODS	DEXA	PAKDD	EDBT	PKDD	WSDM	ECIR	
	#Papers	940	1731	1030	747	680	198	575	...

and effectively integrate heterogeneous information and achieve more reasonable ranks. The detailed method description and validation experiments can been seen in [17].

4.3 Conclusions

Meta path is an unique characteristic of heterogeneous information network. It is an effective semantic capture tool, as well as feature extraction method. As a consequence, meta path play a critical role in data mining tasks on heterogeneous information network. In this chapter, we present two examples on ranking and clustering, respectively. Particularly, we study the ranking problem in heterogeneous information network and propose the HRank framework, which is a path-based random walk method. In addition, we study the ranking-based clustering problem in a general heterogeneous information network and proposed a novel algorithm HeProjI which projects a general HIN with arbitrary schema into a sequence of projected subnetworks and iteratively analyzes each subnetwork. Experiments not only validate their effectiveness but also illustrate the unique advantages of meta path.

Some interesting future works are worth being exploited on meta paths. On the one hand, meta path can be employed on other data mining tasks, so that we can obverse its power and potential on more applications. On the other hand, we need to design more powerful tools, beyond meta path, to capture subtle semantic.

References

1. Brin, S., Page, L.: The anatomy of a large-scale hyper textual web search engine. Comput. Netw. ISDN Syst. **30**(1–7), 1757–1771 (1998)
2. Chen, B., Ding, Y., Wild, D.: Assessing drug target association using semantic linked data. PLoS Comput. Biol. **8**(7)(e1002574), 1757–1771 (2012)
3. Grčar, M., Trdin, N., Lavrač, N.: A methodology for mining document-enriched heterogeneous information networks. Comput. J. **56**(3), 107–121 (2011)
4. Han, J.: Mining heterogeneous information networks by exploring the power of links. In: DS, pp. 13–30 (2009)
5. Jeh, G., Widom, J.: SimRank: a measure of structural-context similarity. In: KDD, pp. 538–543 (2002)
6. Ji, M., Sun, Y., Danilevsky, M., Han, J., Gao, J.: Graph regularized transductive classification on heterogeneous information networks. In: ECML/PKDD, pp. 570–586 (2010)
7. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. In: SODA, pp. 668–677 (1999)
8. Kong, X., Yu, P.S., Ding, Y., Wild, D.J.: Meta path-based collective classification in heterogeneous information networks. In: CIKM, pp. 1567–1571 (2012)
9. Mei, Q., Cai, D., Zhang, D., Zhai, C.: Topic modeling with network regularization. In: WWW, pp. 101–110 (2008)
10. Microsoft: Microsoft Academic. http://academic.research.microsoft.com
11. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E **69**(026113), 1757–1771 (2004)

12. Nie, Z., Zhang, Y., Wen, J.R., Ma, W.Y.: Object-level ranking: bringing order to web objects. In: WWW, pp. 567–574 (2005)
13. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: bringing order to the web. In: Stanford InfoLab, pp. 1–14 (1998)
14. Shi, C., Kong, X., Yu, P.S., Xie, S., Wu, B.: Relevance search in heterogeneous networks. In: International Conference on Extending Database Technology, pp. 180–191 (2012)
15. Shi, C., Yan, Z., Cai, Y., Wu, B.: Multi-objective community detection in complex networks. Appl. Soft Comput. **12**(2), 850–859 (2012)
16. Shi, C., Zhou, C., Kong, X., Yu, P.S., Liu, G., Wang, B.: HeteRecom: a semantic-based recommendation system in heterogeneous networks. In: KDD, pp. 1552–1555 (2012)
17. Shi, C., Wang, R., Li, Y., Yu, P.S., Wu, B.: Ranking-based clustering on general heterogeneous information networks by network projection. In: CIKM, pp. 699–708 (2014)
18. Shi, C., Li, Y., Yu, P.S., Wu, B.: Constrained-meta-path-based ranking in heterogeneous information network. Knowl. Inf. Syst. **49**(2), 1–29 (2016)
19. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **22**(8), 888–905 (2000)
20. Sun, Y., Han, J., Gao, J., Yu, Y.: itopicmodel: information network-integrated topic modeling. In: ICDM, pp. 493–502 (2009)
21. Sun, Y., Han, J., Zhao, P., Yin, Z., Cheng, H., Wu, T.: RankClus: integrating clustering with ranking for heterogeneous information network analysis. In: EDBT, pp. 565–576 (2009)
22. Sun, Y., Yu, Y., Han, J.: Ranking-based clustering of heterogeneous information networks with star network schema. In: KDD, pp. 797–806 (2009)
23. Sun, Y., Norick, B., Han, J., Yan, X., Yu, P.S., Yu, X.: Integrating meta-Path selection with user-guided object clustering in heterogeneous information networks. In: KDD, pp. 1348–1356 (2012)
24. Sun, Y., Norick, B., Han, J., Yan, X., Yu, P.S., Yu, X.: Pathselclus: integrating meta-path selection with user-guided object clustering in heterogeneous information networks. ACM Trans. Knowl. Discov. Data **7**(3), 723–724 (2012)
25. Sun, Y.Z., Han, J.W., Yan, X.F., Yu, P.S., Wu, T.: PathSim: meta path-based top-K similarity search in heterogeneous information networks. In: VLDB, pp. 992–1003 (2011)
26. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: ArnetMiner: extraction and mining of academic social networks. In: KDD, pp. 990–998 (2008)
27. Wang, R., Shi, C., Yu, P.S., Wu, B.: Integrating clustering and ranking on hybrid heterogeneous information network. In: PAKDD, pp. 583–594 (2013)
28. Zhou, D., Orshanskiy, S.A., Zha, H., Giles, C.L.: Co-ranking authors and documents in a heterogeneous network. In: ICDM, pp. 739–744 (2007)