HEFBIB : HIERARCHICAL EXPERT FINDING IN HETEROGENEOUS BIBLIOGRAPHIC NETWORK

BY

HAO LUO

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2015

Urbana, Illinois

Advisor:

Professor Jiawei Han

Abstract

Expert finding systems allow users to type simple text queries and retrieve names of individuals who possess the expertise described in the queries. Such applications are especially useful in real world: conference organizers may search for reviewers, company recruiters may search for talented candidates, graduate students may search for advisers and researchers may search for collaborators, etc. In this study, we propose *Hefbib*, a hierarchical approach to expert finding in heterogeneous bibliographic network, to construct an expert hierarchy given a seed textual topic hierarchy as well as retrieve authoritative experts given a search query. Experiments on synthetic toy examples and real-world DBLP dataset show promising results.

To my family, for their love and support.

Acknowledgments

First and foremost, I would like to express my appreciation to my adviser, Professor Jiawei Han, for his patient guidance and insightful comments on my research work.

I am also very grateful to all the faculties and colleagues in DAIS (Data and Information System) research group at University of Illinois, Urbana-Champaign. I cherish the opportunity to learn from them and gain deeper insights into the field of data mining.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

IR	Information Retrieval
pLSI	Probabilistic Latent Semantic Indexing
LDA	Latent Dirichlet Allocation
MAP	Mean Average Precision
P@K	Precision at Top K Ranked Candidates
NDCG	Normalized Discounted Cumulative Gain

Chapter 1

Introduction

Expert finding systems allow users to type simple text queries and retrieve names of individuals who possess the expertise described in the queries. Such applications are especially useful in real world: conference organizers may search for reviewers, company recruiters may search for talented candidates, graduate students may search for advisers and researchers may search for collaborators, etc.

Expert finding is similar to the traditional ad-hoc information retrieval (IR) tasks since both of them aim at finding the most relevant information given user queries. The major difference is that in the realistic settings of expert finding, the supporting evidences for expertise are not only limited to textual information of documents, but also come from the intersections among heterogeneous entities. Take bibliographic data as an example, since researchers usually publish on various venues (eg., conferences, journals, etc.), collaborate with other researchers and cite other papers, we can obtain the venue information, co-author relationships and citation relationships besides the contents of papers. A lot of previous work [1, 2, 3, 4, 5] have exploited such heterogeneous feature in the expertise network, but none of these models organize the experts into a hierarchy structure with different levels of granularity, which allow users to perform efficient and more meaningful search.

In this study, we propose *Hefbib*, a hierarchical approach to expert finding in heterogeneous bibliographic network, utilizing both topic model and link analysis algorithms. The main contributions of this work are:

- We propose *ExpertFinder*, a generative topic model which utilizes the heterogeneous information in bibliographic network and can recursively construct an expert hierarchy given a seed textual topic hierarchy.
- We propose *BibRank*, a PageRank like ranking algorithm, which allows the authority information to be propagated among heterogeneous entities and hence improve the retrieval results.

The rest of the thesis is organized as follow: In chapter 2, we introduce related concepts and formulate the expert finding problem. In chapter 3, we describe the *ExpertFinder* model and its inference process. The details of *BibRank* ranking algorithm is shown in chapter 4. In chapter 5, we provide experimental

results on real-world dataset and case studies. Chapter 6 introduces state-of-the-art related works. Finally, we conclude the thesis in chapter 7.

Chapter 2

Problem Formulation

In this chapter, we introduce related concepts with notations, and formally define the problem of expert finding in heterogeneous bibliographic networks.

2.1 Related Concepts

Definition 2.1. (Information Network). An information network consists of T types of objects $\mathcal{X} = \{X_t\}_{t=1}^{T}$, where X_t is a set of objects belonging to type t. Such a network can be denoted as a weighted graph $G = \langle \mathcal{X}, E, W \rangle$, where $\mathcal{X}$ is a set of vertices representing different types of objects, E is a set of edges representing the binary relation between objects and $W : E \to \mathbb{R}^+$ is a set of weights mapping from an edge $e \in E$ to a real number $w \in \mathbb{R}^+$. Specifically, the network is called **heterogeneous information network** when $T \geq 2$; and **homogeneous information network** when $T = 1$.

Definition 2.2. (Heterogeneous Bibliographic Network). A heterogeneous bibliographic network is a special kind of heterogeneous information network, consisting objects of types **Author**, **Paper**, **Venue** and **Term**, etc. A toy example of DBLP bibliographic network is shown in 2.1.

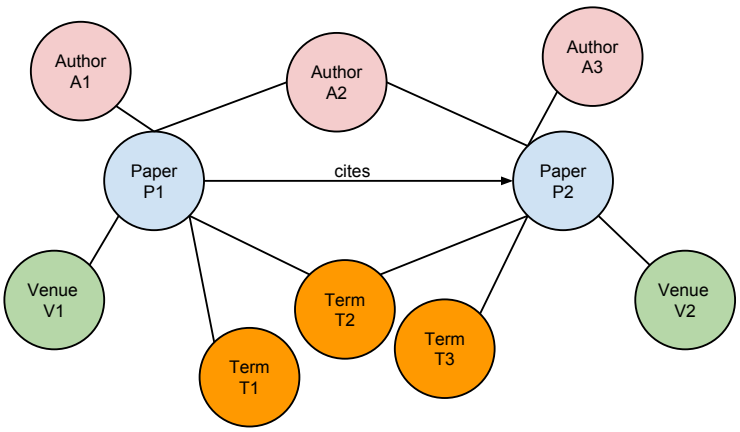


Figure 2.1: A toy example of DBLP bibliographic network.

Definition 2.3. (Topical Hierarchy). A topical hierarchy [6] can be defined as a tree $\mathcal{T}$ in which each node is a topic. The root topic is denoted as o. Every non-root topic t with parent topic $par(t)$ is represented by a ranked list of phrases $\{P^t, r^t(P^t)\}$, where P^t is the set of phrases for topic t and $r^t(P^t)$ is the ranking scores for the phrases in topic t. For every non-leaf topic t in the tree, all of its subtopics comprise its children set $C^t = \{z \in \mathcal{T}, par(z) = t\}$. A phrase can appear in multiple topics, though it may have a different ranking score in each topic. An example topical hierarchy of computer science research areas is shown in 2.2

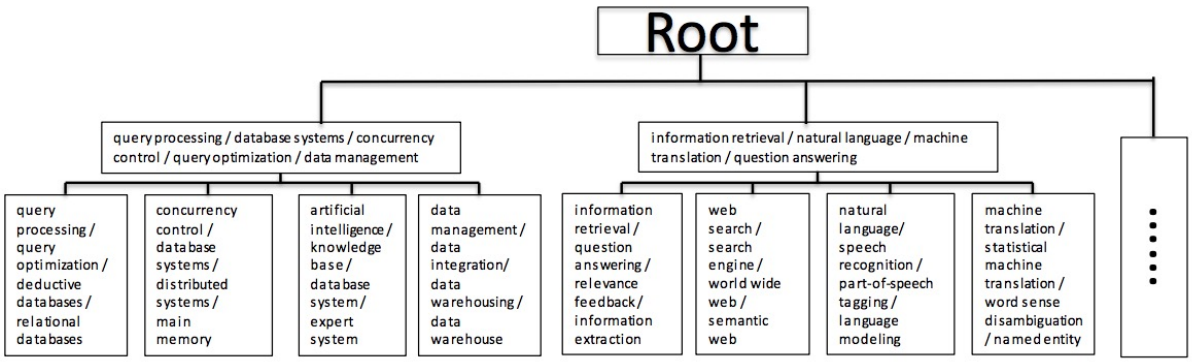


Figure 2.2: Topical hierarchy for computer science research areas.

Definition 2.4. (Topical Expert Hierarchy). A topical expert hierarchy $\mathcal{E}$ also maintains a tree structure in which each node is a topic. It can be constructed from a given topical hierarchy $\mathcal{T}$ with the same topic distributions. Every non-root topic t with parent topic $par(t)$ is represented by a ranked list of authors $\{A^t, r^t(A^t)\}$, where A^t is the set of experts for topic t and $r^t(A^t)$ is the ranking scores for the experts in topic t.

2.2 Problem Statement

Now we can formulate the hierarchical expert finding problem as follow: given a concept hierarchy $\mathcal{T}$ and a heterogeneous bibliographic network G, construct an expert hierarchy $\mathcal{E}$, such that for all topics t, A^t represents a rank list of experts for the topic that compromises phrases P^t.

2.3 General Framework

Figure 2.3 presents the framework of *HefBib*. It consists of two major steps: (1) Offline topical expert hierarchy construction and (2) Online query searching. For the offline step, a topical expert hierarchy is constructed given the topical hierarchy and the heterogeneous bibliographic network. For each topic in

the hierarchy, ranking distributions of authors and venues are firstly inferred from a probabilistic generative model. Authors with top ranking scores are output as expert candidates, which guarantees that all candidates are related to the topic. Intuitively, an expert should also be authoritative besides relevant. Hence the authority scores of heterogeneous entities are propagated within the network. We summarize the framework of *HefBib* as follow:

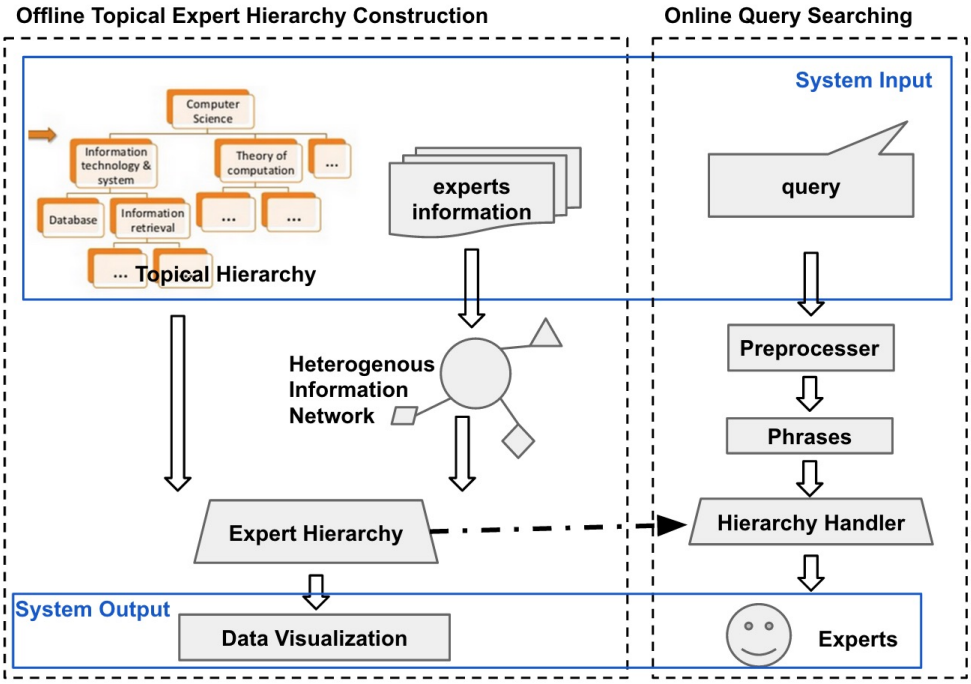


Figure 2.3: System framework of *Hefbib*

1 Offline Topical Expert Hierarchy Construction

(a) Preprocessing: extend the input topical hierarchy with a few seed phrases to a more complete topical hierarchy with abundant phrases and quantitative ranking scores.

(b) Construct topical expert hierarchy:

i. Build a novel probabilistic generative model *ExpertFinder* for the heterogeneous bibliographic network and infer ranking distributions of authors and venues for each topic on the current level.

ii. For each topic on the current level, propagate the topical ranking scores of authors and venues with a novel ranking algorithm *BibRank*.

iii. Recursively apply steps b(i) - b(ii) to each topic to construct the topical expert hierarchy in a top-down fashion.

2 Online Query Searching

(a) Segment the input query into key phrases.

(b) Locate each key phrase at the most "specific" level in the expert hierarchy and obtain the corresponding experts.

(c) Combine and re-rank the experts.

The thesis mainly addresses the offline topical expert hierarchy construction part and will leave the online query searching as future work.

Chapter 3

ExpertFinder: Generative Model for Heterogeneous Topic Modeling

In this chapter, we introduce a generative model, *ExpertFinder*, which utilizes the phrase ranking distribution in pre-defined topical hierarchy to do topic modeling in heterogeneous bibliographic network.

3.1 Generative process

In *ExpertFinder* model (Figure 3.1), we represent each paper as a *bag of phrases, authors* and *venue*. Each paper has a distribution over topics and each topic has a distribution over phrases, authors and venues. The intuition behind this model is: when writing a paper, the coauthors, contents and the publication venues are chosen based on the research topic. The corresponding generative process can be summarized as follow:

1. For each topic distribution:
   (a) Draw the topic distribution $\theta \sim Dirichlet(\alpha)$, where α is a Dirichlet prior.
2. For each topic $z = 1...K$:
   (a) Draw the author distribution $\phi_z \sim Dirichlet(\beta)$, where β is a Dirichlet prior.
   (b) Draw the venue distribution $\varphi_z \sim Dirichlet(\gamma_z)$, where γ_z are Dirichlet priors.
3. For each paper $d \in D$
   (a) Draw a topic $z \sim Categorical(\theta)$
   (b) For each phrase p of paper: draw a phrase $p \sim Categorical(\eta_z)$
   (c) For each author a of paper: draw an author $a \sim Categorical(\phi_z)$
   (d) For the paper venue v: draw a venue $v \sim Categorical(\varphi_z)$

The notations are shown in Table 3.1.

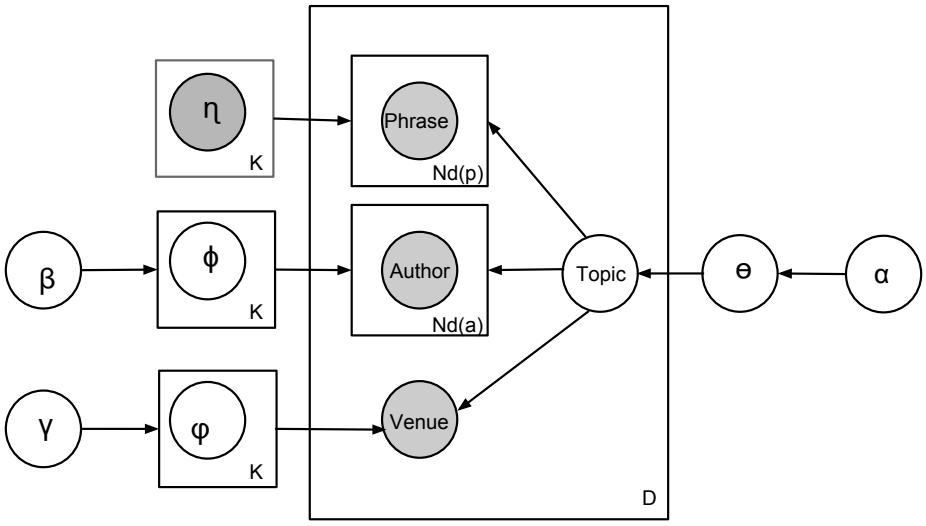


Figure 3.1: Plate representation of *ExpertFinder* model

Symbol	Description
$\boldsymbol{p}$	observation of phrases among all papers
$\boldsymbol{a}$	observation of authors among all papers
$\boldsymbol{a}$	observation of venues among all papers
$\boldsymbol{z}$	latent topic assignments of all papers
$\boldsymbol{\theta}$	latent topic distribution
$\boldsymbol{\phi}$	latent author ranking distribution under all subtopics
$\boldsymbol{\varphi}$	latent venue ranking distribution under all subtopics
$\boldsymbol{\eta}$	given phrase ranking distribution under all subtopics
α, β, γ	hyperparameters of $\boldsymbol{\theta}$, $\boldsymbol{\phi}$ and $\boldsymbol{\varphi}$
N_z^D	the number of papers assigned topic z
$N_{z,a}^A$	the number of papers author a publish assigned topic z
$N_{z,p}^P$	the number of papers phrase p appear assigned topic z
$N_{z,v}^V$	the number of papers venue v accept assigned topic z
A_i	authors of paper d_i
P_i	phrases of paper d_i
V_i	venue of paper d_i
$N_{i,p}$	the number of phrase p appear in paper d_i

Table 3.1: Notations of the *ExpertFinder* model.

3.2 Model Inference

Based on *ExpertFinder's* generative process, the joint distribution of all random variables can be derived as:

$$P(\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{v}, \boldsymbol{z}, \boldsymbol{\phi}, \boldsymbol{\varphi}, \boldsymbol{\theta}; \alpha, \beta, \gamma, \boldsymbol{\eta}) = P(\boldsymbol{\theta}|\alpha)P(\boldsymbol{\phi}|\beta)P(\boldsymbol{\varphi}|\gamma)P(\boldsymbol{z}|\boldsymbol{\theta})P(\boldsymbol{p}|\boldsymbol{\eta}, \boldsymbol{z})P(\boldsymbol{a}|\boldsymbol{\phi}, \boldsymbol{z})P(\boldsymbol{v}|\boldsymbol{\varphi}, \boldsymbol{z}) \tag{3.1}$$

Since we do not care about the topic distributions over all papers, the joint distribution in equation 3.1 can be rewritten by integrating out θ as:

$$\begin{aligned} P(\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{v}, \boldsymbol{z}, \boldsymbol{\phi}, \boldsymbol{\varphi}; \alpha, \beta, \gamma, \boldsymbol{\eta}) &= \int P(\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{v}, \boldsymbol{z}, \boldsymbol{\phi}, \boldsymbol{\varphi}, \boldsymbol{\theta}; \alpha, \beta, \gamma, \boldsymbol{\eta}) d\boldsymbol{\theta} \\ &= P(\boldsymbol{\phi}|\beta)P(\boldsymbol{a}|\boldsymbol{\phi}, \boldsymbol{z})P(\boldsymbol{\varphi}|\gamma)P(\boldsymbol{v}|\boldsymbol{\varphi}, \boldsymbol{z})P(\boldsymbol{p}|\boldsymbol{\eta}, \boldsymbol{z}) \int P(\boldsymbol{\theta}|\alpha)P(\boldsymbol{z}|\boldsymbol{\theta}) d\boldsymbol{\theta} \end{aligned} \tag{3.2}$$

where,

$$\begin{aligned} P(\boldsymbol{\phi}|\beta)P(\boldsymbol{a}|\boldsymbol{\phi}, \boldsymbol{z}) &= \prod_{z=1}^{K} P(\phi_z|\beta)P(\boldsymbol{a}|\phi_z) \\ &= \prod_{z=1}^{K} \frac{1}{B(\beta)} \prod_{a=1}^{A} \phi_{z,a}^{\beta_a - 1} \prod_{a=1}^{A} \phi_{z,a}^{N_{z,a}^{A}} \\ &= \prod_{z=1}^{K} \frac{1}{B(\beta)} \prod_{a=1}^{A} \phi_{z,a}^{N_{z,a}^{A} + \beta_a - 1} \end{aligned} \tag{3.3}$$

$$P(\boldsymbol{\varphi}|\gamma)P(\boldsymbol{v}|\boldsymbol{\varphi}, \boldsymbol{z}) = \prod_{z=1}^{K} \frac{1}{B(\gamma)} \prod_{v=1}^{V} \varphi_{z,v}^{N_{z,v}^{V} + \gamma_v - 1} \tag{3.4}$$

$$P(\boldsymbol{p}|\boldsymbol{\eta}, \boldsymbol{z}) = \prod_{z=1}^{K} \prod_{p=1}^{P} \eta_{z,p}^{N_{z,p}^{P}} \tag{3.5}$$

$$\begin{aligned} \int P(\boldsymbol{\theta}|\alpha)P(\boldsymbol{z}|\boldsymbol{\theta}) d\boldsymbol{\theta} &= \int \frac{1}{B(\alpha)} \prod_{z=1}^{K} \theta_z^{\alpha_z - 1} \prod_{z=1}^{K} \theta_z^{N_z^D} d\theta_z \\ &= \frac{1}{B(\alpha)} \int \prod_{z=1}^{K} \theta_z^{N_z^D + \alpha_z - 1} d\theta_z \\ &= \frac{B(N^D, . + \alpha)}{B(\alpha)} \end{aligned} \tag{3.6}$$

Bringing equations 3.3, 3.4, 3.5 and 3.6 back to 3.2, we have:

$$P(\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{v}, \boldsymbol{z}, \boldsymbol{\phi}, \boldsymbol{\varphi}; \alpha, \beta, \gamma, \boldsymbol{\eta}) = \frac{B(N^D, . + \alpha)}{B(\alpha)} \prod_{z=1}^{K} \frac{\prod_{a=1}^{A} \phi_{z,a}^{N_{z,a}^{A} + \beta_a - 1} \prod_{v=1}^{V} \varphi_{z,v}^{N_{z,v}^{V} + \gamma_v - 1} \prod_{p=1}^{P} \eta_{z,p}^{N_{z,p}^{P}}}{B(\beta)B(\gamma)} \tag{3.7}$$

We use collapsed Gibbs sampling to infer the model. The basic idea in Gibbs sampling is that, rather

than probabilistically picking the next state all at once, we make a separate probabilistic choice for each dimension, where each choice depends on the other dimensions.

1. **Sampling for paper topic label**

   Let $\mu = \alpha, \beta, \gamma, \boldsymbol{\eta}$, we can obtain the distribution we are sampling from, the posterior probability of latent topic label for each paper d_i in our case, by using the definition of conditional probability:

$$\begin{aligned} P(z_i|\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{v}, \boldsymbol{z}^{(-i)}, \boldsymbol{\phi}, \boldsymbol{\varphi}; \mu) &= \frac{P(\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{v}, z_i, \boldsymbol{z}^{(-i)}, \boldsymbol{\phi}, \boldsymbol{\varphi}; \mu)}{P(\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{v}, \boldsymbol{z}^{(-i)}, \boldsymbol{\phi}, \boldsymbol{\varphi}; \mu)} \\ &= \frac{P(\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{v}, \boldsymbol{z}, \boldsymbol{\phi}, \boldsymbol{\varphi}; \mu)}{P(\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{v}, \boldsymbol{z}^{(-i)}, \boldsymbol{\phi}, \boldsymbol{\varphi}; \mu)} \\ &= \frac{B(N^D, .+\alpha)}{B(N^{D^{(-i)}}, .+\alpha)} \prod_{z=1}^{K} (\prod_{a=1}^{A} \frac{\phi_{z,a}^{N_{z,a}^A + \beta_a - 1}}{\phi_{z,a}^{N_{z,a}^{A^{(-i)}} + \beta_a - 1}} \prod_{v=1}^{V} \frac{\varphi_{z,v}^{N_{z,v}^V + \gamma_v - 1}}{\varphi_{z,v}^{N_{z,v}^{V^{(-i)}} + \gamma_v - 1}} \prod_{p=1}^{P} \frac{\eta_{z,p}^{N_{z,p}^P}}{\eta_{z,p}^{N_{z,p}^{P^{(-i)}}}}) \end{aligned} \tag{3.8}$$

   where

$$\begin{aligned} \frac{B(N^D, .+\alpha)}{B(N^{D^{(-i)}}, .+\alpha)} &= \frac{\prod_{z=1}^{K} \Gamma(N_z^d + \alpha_z)}{\prod_{z=1}^{K} \Gamma(N_z^{d^{(-i)}} + \alpha_z)} \frac{\Gamma(\sum_{z=1}^{K} N_z^{d^{(-i)}} + \alpha_z)}{\Gamma(\sum_{z=1}^{K} N_z^d + \alpha_z)} \\ &= \frac{N_{z_i}^{d^{(-i)}} + \alpha_{z_i}}{\Gamma(\sum_{z=1}^{K} N_z^{d^{(-i)}} + \alpha_z)} \\ &\propto N_{z_i}^{d^{(-i)}} + \alpha_{z_i} \end{aligned} \tag{3.9}$$

$$\prod_{z=1}^{K} \prod_{a=1}^{A} \frac{\phi_{z,a}^{N_{z,a}^A + \beta_a - 1}}{\phi_{z,a}^{N_{z,a}^{A^{(-i)}} + \beta_a - 1}} = \prod_{a \in A_i} \phi_{z_i,a} \tag{3.10}$$

$$\prod_{k=1}^{K} \prod_{v=1}^{V} \frac{\varphi_{z,v}^{N_{z,v}^V + \gamma_v - 1}}{\varphi_{z,v}^{N_{z,v}^{V^{(-i)}} + \gamma_v - 1}} = \varphi_{z_i,v_i} \tag{3.11}$$

$$\prod_{k=1}^{K} \prod_{p=1}^{P} \frac{\eta_{z,p}^{N_{z,p}^P}}{\eta_{z,p}^{N_{z,p}^{P^{(-i)}}}} = \prod_{p \in P_i} \eta_{z_i,p}^{N_{i,p}} \tag{3.12}$$

   Bringing equations 3.9, 3.10, 3.11 and 3.12 back to equation 3.8, we have:

$$P(z_i|\boldsymbol{p}, \boldsymbol{a}, \boldsymbol{v}, \boldsymbol{z}^{(-i)}, \boldsymbol{\phi}, \boldsymbol{\varphi}; \mu) \propto (N_{z_i}^{D^{(-i)}} + \alpha_{z_i}) \prod_{a \in A_i} \phi_{z_i,a} \times \varphi_{z_i,v_i} \times \prod_{p \in P_i} \eta_{z_i,p}^{N_{i,p}} \tag{3.13}$$

2. **Sampling for author and venue topical ranking distribution**

   Since $\boldsymbol{\phi}$ and $\boldsymbol{\varphi}$ are both conjugate priors, whose posterior, like the prior, works out to be Dirichlet distribution, hence we could sample the new values by making another draw from the Dirichlet

distribution with parameters $N_z^A, . + \beta$ and $N_z^V, . + \gamma$:

$$\begin{cases} \phi_z \sim Dirichlet(N_z^A, . + \beta) \\ \varphi_z \sim Dirichlet(N_z^V, . + \gamma) \end{cases} \tag{3.14}$$

We summarize the whole process of Gibbs sampling in Algorithm 1.

Algorithm 1 Gibbs sampling for *ExpertFinder*

```
Initialized related parameters.
for t := 1 to T do
    for i := 1 to |D| do
        Subtract 1 from the count of papers with topic label z_i
        for a ∈ A_i do
            Subtract 1 from author a's count of papers with topic label z_i
        end for
        Subtract 1 from venue V_i's count of papers with topic label z_i
        Assign a new topic label z_i^(t+1) to paper d_i as described in Equation 3.13.
        Add 1 to the count of papers with topic label z_i^(t+1)
        for a ∈ A_i do
            add 1 to author a's count of papers with topic label z_i^(t+1)
        end for
        Add 1 to venue V_i's count of papers with topic label z_i^(t+1)
    end for
    φ_z ~ Dirichlet(N_z^A, . + β)
    ϕ_z ~ Dirichlet(N_z^V, . + γ)
end for
```

Chapter 4

BibRank: Authority Propagation in Heterogeneous Bibliographic Network

In this chapter, we introduce a novel link analysis and authority propagation model *BibRank*.

4.1 Motivation

As discussed previously, an author can be taken as an "expert" regarded to a specific topic should at least have two properties: 1) high relevancy to the topic 2) high authority within the topic. The *ExpertFinder* topic model described in chapter 3 mainly addresses the first property while does not take into account the authority information. Let's imagine a scenario where two authors a_1 and a_2 publish nearly the same contents. a_1 mostly publish in top venues while a_2 mostly publish in junk venues. Undoubtedly, a_1 should be taken as an "expert" while a_2 should not. If we simply adopt *ExpertFinder*, the topical ranking scores of a_1 and a_2 will be very similar since the model ranks the entities with text information (phrases distribution in our case). Hence we develop a ranking mechanism , called *BibRank*, to "kick out" the false positives and mine the real experts.

4.2 Intuition

Following section 4.1, the intuitions of *BibRank* model could be derived from the perspectives of two kinds of errors: false positives and false negatives. The false positives are referred to the authors who are taken as experts but are not actually qualified. The false negatives are those who should have been taken as experts but are underrated. We summarize their properties respectively as follow:

- False positives (unqualified experts)
  - Publishes many low quality papers in junk venues.
  - Does not contribute much effort to a paper, just adding the name to the author list.
- False negatives (underrated experts)

– Young researches who have not accumulated large number of citations.

Besides, we also want to incorporate the following intuitions:

- Authors who cite their own papers a lot should be detected and penalized to some extent.
- Papers cited by high quality papers should have higher authority scores than those cited by mediocre or junk papers. (PageRank philosophy)

4.3 Model Description

In this section, we describe the details of *BibRank* ranking model and illustrate how it can capture the intuitions in the previous section.

Similarly to traditional PageRank algorithm, *BibRank* also models a random walk process. The "random surfer" starts the random walk on the heterogeneous bibliographic network following the entity relationship links, never hitting back but eventually gets bored and will restart his random walk on the network again to find another seed object.

We use a vector $\mathbb{I}_{\boldsymbol{X,z}}$ to denote the initial topical ranking score of the entity of type X, which is the probability that the "random surfer" finds the entity only based on the topical relevancy. This value could be either direct or post-processed output of *ExpertFinder* model. We use another vector $\boldsymbol{R}_{\boldsymbol{X,z}}$ to denote the probability that he finds the entity through the relationship links. To compute the authority score of an entity, the *BibRank* model takes into account both the topical relevancy and its relationships with other entities in the network. For example, the ranking scores of junk venues will be propagated to the authors who have publications on it and hence penalize the ranking scores of these authors. In this way, authors who have high quality publications can be differentiated from those who only have low quality publications.

In summary, a highly ranked paper should be written by authoritative authors, published on top venues and is cited frequently (by good papers); a highly ranked author should publish many high quality papers and collaborate with good authors; and a highly ranked venue should attract many good papers. We use the following formulas to represent these authority propagation relationships:

$$\begin{cases} \boldsymbol{R}_{\boldsymbol{D,z}}^{(t+1)} = \varepsilon_D \mathbb{I}_{\boldsymbol{D,z}}^{(t)} + (1-\varepsilon_D)(\gamma_{DA}\boldsymbol{M_{DA}}\boldsymbol{R}_{\boldsymbol{A,z}}^{(t)} + \gamma_{DV}\boldsymbol{M_{DV}}\boldsymbol{R}_{\boldsymbol{V,z}}^{(t)} + \gamma_{DD}\boldsymbol{M_{DD}^T}\boldsymbol{R}_{\boldsymbol{D,z}}^{(t)}) \\ \boldsymbol{R}_{\boldsymbol{A,z}}^{(t+1)} = \varepsilon_A \mathbb{I}_{\boldsymbol{A,z}}^{(t)} + (1-\varepsilon_A)(\gamma_{AD}\boldsymbol{M_{AD}}\boldsymbol{R}_{\boldsymbol{D,z}}^{(t)} + \gamma_{AA}\boldsymbol{M_{AA}}\boldsymbol{R}_{\boldsymbol{A,z}}^{(t)}) \\ \boldsymbol{R}_{\boldsymbol{V,z}}^{(t+1)} = \varepsilon_V \mathbb{I}_{\boldsymbol{V,z}}^{(t)} + (1-\varepsilon_V)\boldsymbol{M_{DV}^T}\boldsymbol{R}_{\boldsymbol{D,z}}^{(t)} \end{cases} \tag{4.1}$$

where

- z: topic label
- $\boldsymbol{R_{D,z}}$, $\boldsymbol{R_{A,z}}$, $\boldsymbol{R_{V,z}}$: vector of authority scores of papers, authors and venues under topic z;
- $\mathbb{I}_{\boldsymbol{D,z}}$, $\mathbb{I}_{\boldsymbol{A,z}}$, $\mathbb{I}_{\boldsymbol{V,z}}$: initial authority scores of papers, authors and venues under topic z, inferred from the *ExpertFinder* generative model.
- ε_X ($X \in \{D, A, V\}$): damping factor which is the probability that random surfer starts with the initial authority score.
- γ_{YX} ($X, Y \in \{D, A, V\}$): authority propagation factor of relationship links from entities of type Y to entities of type X, and $\sum_{\forall Y} \gamma_{YX} = 1$
- $\boldsymbol{M_{XY}}$: adjacency matrix between entities of type X and entities of type Y. Specifically:
  - $\boldsymbol{M_{DA}}$: normalized adjacency matrix of paper to author, eg. $m_{da} = \frac{w_{ad}}{\sum_{d' \in D_a} w_{ad'}}$, where w_{ad} denotes the "effort" paper d received from author a. This could be assigned according to the order of author (eg. if an author is the first author of one paper and the last author of another paper, we could assume that the author contribute more to the former).
  - $\boldsymbol{M_{AD}}$: normalized adjacency matrix of author and paper, eg. $m_{ad} = \frac{w_{da}}{\sum_{a' \in A_d} w_{da'}}$, where w_{da} denotes the "importance" of author a regarded to paper d. This could be assigned according to the order of author (eg. first author usually contribute more compared to others).

    Note: Notice that here matrix $\boldsymbol{M_{AD}}$ is **not** the transpose of matrix $\boldsymbol{M_{DA}}$. Suppose there three papers as follow: Then $\boldsymbol{M_{AD}}$ and $\boldsymbol{M_{DA}}$ should look like as follow:

	a_1	a_2	a_3
d_1	0	1	1
d_2	1	1	0
d_3	0	0	1

Table 4.1: Boolean adjacency matrix

	a_1	a_2	a_3
d_1	0	$\frac{2}{3}$	$\frac{1}{3}$
d_2	1	$\frac{1}{3}$	0
d_3	0	0	$\frac{2}{3}$

	d_1	d_2	d_3
a_1	0	$\frac{2}{3}$	0
a_2	$\frac{2}{3}$	$\frac{1}{3}$	0
a_3	$\frac{1}{3}$	0	1

Table 4.2: $\boldsymbol{M_{DA}}$(left) and $\boldsymbol{M_{AD}}$ (right), each column sums up to 1.

  - $\boldsymbol{M_{DV}}$: adjacency matrix of paper to venue.

– $\boldsymbol{M_{DD}}$: adjacency matrix of paper to the papers that cites it.

– $\boldsymbol{M_{AA}}$: normalized adjacency matrix of co-author relationships. The association strength between two authors are determined on the base of two factors: 1) frequency of co-authorship and 2) total number of co-authors on papers. Suppose author a and author a' collaborates on paper d, then we could define the *exclusivity* $g_{a,a',d}$ as follow:

$$g_{a,a',d} = \frac{1}{|A_d|} \tag{4.2}$$

where $|A_d|$ denotes the total number of authors of paper d. The fewer authors a paper has, the higher association each pair of its co-authors has.

Then we define the *topical co-authorship frequency* $w_{a,a',z}$ as follow:

$$w_{a,a',z} = \sum_d g_{a,a',d} \tag{4.3}$$

Then for $m_{aa'} \in M_{AA}$, we have $m_{aa'} = \frac{w_{a,a',z}}{\sum_{a'' \in A} w_{a,a'',z}}$

Chapter 5

Experiments

In this chapter, we evaluate the effectiveness of our *HefBib* model, and compare it with the state-of-the-art methods on DBLP datasets through extensive experiments.

5.1 Data Preparation

5.1.1 Datasets

DBLP is a collection of bibliographic information on major computer science journals and proceedings, which can be used to build a heterogeneous information network with multi-typed (paper, venue, term, author, etc.) objects. Tang *et al.* [5] extracted meta information of papers and built a DBLP dataset. In this experiment, we use a subset this DBLP dataset that belong to four research areas: *data mining*, *database*, *information retrieval* and *machine learning*. Statistics of the heterogeneous bibliographic network constructed from DBLP dataset are summarized in Table 5.1 and Table 5.2.

# of papers	34656
# of authors	38491
# of venues	23
# of phrases	5100

Table 5.1: Statistics of DBLP dataset.

Research area	Venues
Data mining	KDD, ICDE, CIKM, WSDM, ICDM, PKDD, PAKDD, TKDE
Machine learning	AAAI, AJCAI, UAI, ICML, ECML,
Database	VLDB, SIGMOD, ICDT, EDBT,
Information retrieval	Conference on Recommender Systems, SIGIR, JCDL, ECDL, ECIR

Table 5.2: Selected venues of DBLP dataset.

5.1.2 Preprocessing

Extend *Seed* Topical Hierarchy

Quite a few recent works [6] have tackled the challenge of automatically constructing a topical hierarchy with quantitative phrase ranking scores and they can certainly be taken as the input topical hierarchy of our *HefBib* model.

On the other hand, a topical hierarchy can also be explicitly specified by domain experts, which introduces human guidance and hence is more accurate. In real world scenarios, however, it is unlikely for domain experts to provide a complete phrase list with ranking scores since it costs too much human efforts. Instead, domain experts can just specify a few "seed" phrases for each topic in the hierarchy structure. For example, a researcher in machine learning area may specify {"machine learning", "supervised learning", "unsupervised learning", "active learning", "reinforcement learning"} for the topic of "machine learning" and then specify {"supervised learning", "classification", "regression", "support vector machine"} for the sub-topic of "supervised learning". Then our first preprocessing task is to extend this "seed" topical hierarchy to a more complete topical hierarchy with quantitative phrase ranking scores, just as the output of CATHY [6].

This task itself has been a challenging research topic and is out of the scope of our *HefBib* model. Hence we use a simple data driven approach to tackle this problem, as is summarized as follow:

1 Train a word2vec[1] model with all supporting documents, e.g, paper titles and abstracts.

2 For each topic t, domain experts specify seed phrases P^t_{seed} with corresponding ranking scores R^t_{seed}.

3 For each seed phrase $p^t_{seed} \in P^t_{seed}$, obtain semantically similar phrases p^t_{sim} with corresponding similarity scores r^t_{sim} from word2vec model. Then for each similar phrase $p^t_{sim_i} \in p^t_{sim}$, the final topical ranking scores is $r^t_{sim_i} = r^t_{sim_i} r^t_{seed}$

4 Hence we map each topic t to a phrase ranking list $\{P^t_{sim}, R^t_{sim}\}$, where P^t_{sim} is the key phrases and R^t_{sim} is the corresponding topical ranking scores.

Figure 5.1 presents an example of extending *machine learning* sub-topic.

Bag-of-Phrases Extraction

The *bag-of-words* model is a simplifying representation used in natural language processing and information retrieval (IR). In this model, a text (such as a sentence or a document) is represented as the bag (multiset) of its words, disregarding grammar and word order but keeping multiplicity. In this experiment, we use its extension, *bag-of-phrases*, to represent text information of papers. As the name implies, the *bag-of-phrases*

[1] https://code.google.com/p/word2vec/

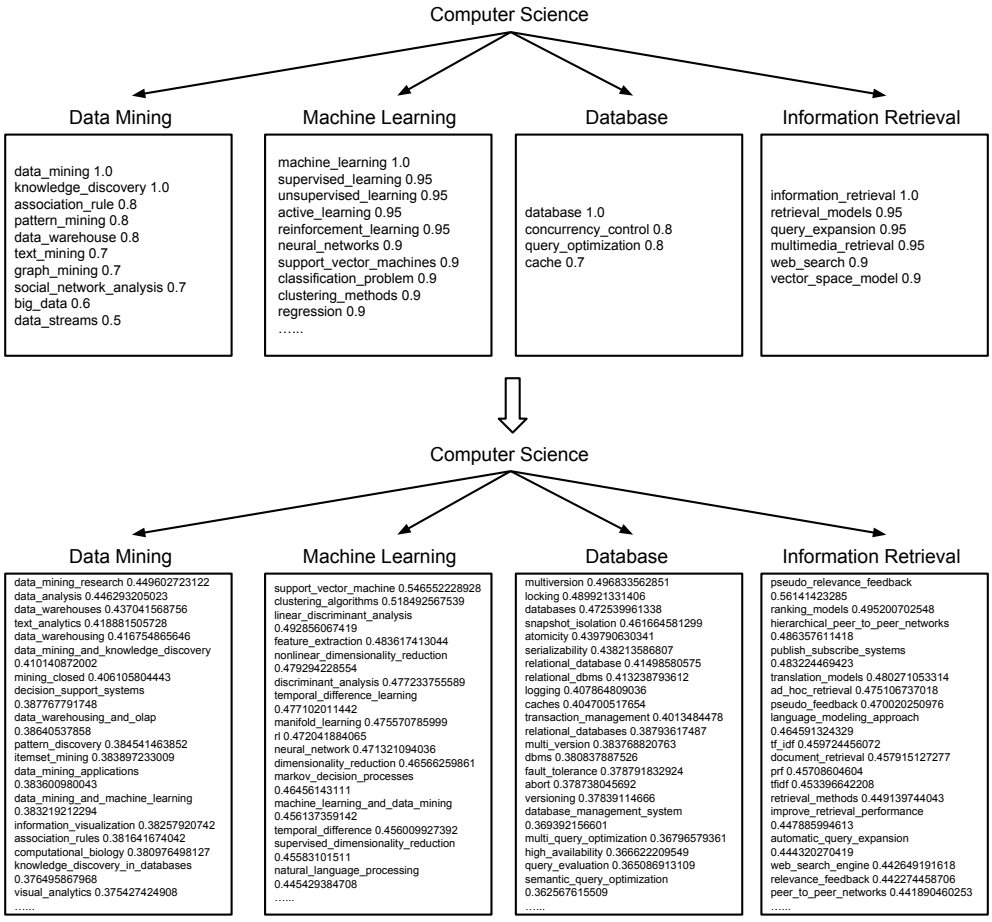


Figure 5.1: Extend *seed* topical hierarchy.

model transfers text data from word granularity to phrase granularity, which reduces semantic ambiguity and hence enhances the power and efficiency at manipulating unstructured data.

In our experiments, we extract key phrases from titles and abstracts of papers using a recently developed phrase mining approach *SegPhrase+* [7].

5.2 Experimental Settings

We provide details on the experimental settings for conducting evaluations on all the methods.

5.2.1 Evaluation Metrics

For the evaluation, we want to assess the ability of our model to construct an expert hierarchy that human judgment deems to be of high quality.

Intrusion Test

Similar to CATHYHIN[8], we adapt the task from Chang *et al.* [9], who were the first to explore human evaluation of topic models. The task involves a set of questions asking humans to discover the "intruder" entity from several options. The evaluation scores were pools for all annotators. The first task is *Author Intrusion*, which evaluates how well the expert hierarchy can separate authors in the dataset into different topics. Each question consists of X authors, $X-1$ of them are randomly chosen from the top authors of the same topic and the remaining author is randomly chosen from a sibling topic. The second task is *Venue Intrusion*, which is similar to the first task except that we evaluate venues instead of authors.

P@k and NDCG

Besides, we also evaluate the results based on calculating popular Information Retrieval (IR) performance metrics: Precision at top k ranked candidate experts (P@k) (k = 5 in our case) and Normalized Discounted Cumulative Gain (NDCG).

Specifically, for modern information retrieval, recall is no longer a meaningful metric, as many queries have thousands of relevant documents, and few users will be interested in reading all of them. For example, in our expert finding systems, there are usually a large number of relevant experts given a query and it is difficult and not necessary to find all of them. Hence the P@k metric allow us to evaluate the system based on the results on the "first page" the search engine retrieves. NDCG uses a graded relevance scale of documents from the result set to evaluate the usefulness, or gain, of a document based on its position in the

result list. The premise of NDCG is that highly relevant documents appearing lower in a search result list should be penalized as the graded relevance value is reduced logarithmically proportional to the position of the result. The DCG accumulated at a particular rank position p is defined as:

$$\mathrm{DCG_p} = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i}. \tag{5.1}$$

Since result set may vary in size among different queries or systems, to compare performances the normalized version of DCG uses an ideal DCG. To this end, it sorts documents of a result list by relevance, producing an ideal DCG at position p(IDCG_p), which normalizes the score:

$$\mathrm{NDCG_p} = \frac{\mathrm{DCG_p}}{\mathrm{IDCGp}} \tag{5.2}$$

5.2.2 Comparing Methods

We compared the proposed method *HefBib* and its variation(*HefBib-*) with several state-of-the-art approaches.

- **HefBib**: Use a predefined topical hierarchy (with phrase ranking distribution under each sub-topic).
- **ExpertFinder**: The first part of *Hefbib*, without propagating authority information among entities.
- **HefBib-**: Infers the phrase ranking distribution automatically without a predefined topical hierarchy.
- **CATHYHIN**: The current state-of-the-art topical hierarchy construction method in heterogeneous information network proposed by Wang *et al.*[8].

5.3 Experiment Results

Intruder Detection

	Author	Venue
HefBib	**0.75**	**1.0**
ExpertFinder	0.38	0.75
HefBib-	0.38	0.58
CATHYHIN	0.33	0.55

Table 5.3: Results of intruder detection.

P@k and NDCG

K = 5 in our experiments.

	P@5	NDCG
HefBib	**0.92**	**0.84**
ExpertFinder	0.89	0.78
HefBib-	0.62	0.53
CATHYHIN	0.72	0.61

Table 5.4: Results of P@5 and NDCG.

5.4 Case Study

5.4.1 DBLP dataset

Let's examine the *ExpertFinder* model by looking into the *data mining* sub-topic. The top authors and venues are shown in 5.5.

ExpertFinder	*HefBib*
Jiawei Han	Jiawei Han
Philip S. Yu	Philip S. Yu
Wei Wang	Christos Faloutsos
Christos Faloutsos	Charu C. Aggarwal
Qiang Yang	Qiang Yang
Jian Pei	Ming-Syan Chen
Hans-Peter Kriegel	Jian Pei
Ming-Syan Chen	Wei Wang
Haixun Wang	Bing Liu
Tao Li	Hans-Peter Kriegel
Jeffrey Xu Yu	Tao Li
Charu C. Aggarwal	Xindong Wu
Zhi-hua Zhou	Zhi-hua Zhou
Ke Wang	Eamonn Keogh
Jieping Ye	Ke Wang

(a) Authors

ExpertFinder	*HefBib*
KDD	KDD
IDCM	ICDM
PAKDD	PAKDD
ICML	TKDE
TKDE	CIKM
CIKM	IJCAI
PKDD	ECMLPKDD
ECMLPKDD	SIGMOD
IJCAI	PKDD
ECML	ICDE

(b) Venues

Table 5.5: Top authors and venues in *data mining*.

As we can see, *ExpertFinder* outputs promising results while *HefBib* which incorporates authority propagation improves the results further. For example, Charu C. Aggarwal is a prestigious researcher in data mining and his ranking is improved with *HefBib* since he usually publishes on good venues and his papers are frequently cited. Another example is the venue ranking. The ACM SIGMOD conference is not included

in the top ten venues output by *ExpertFinder* but is included by *HefBib*. This is because ACM SIGMOD conference has attracted many good authors publishing data mining papers during recent years.

5.4.2 Synthetic toy dataset

To prove the effectiveness of our model, I designed a toy dataset. Suppose there are seven authors as follow (the first six authors are all highly regarded to a specific topic while the last one is not):

id	description	feature
a_0	Publishes ten papers on good venues as the first author (co-author with a_2); accumulates high citations	Prestigious expert
a_1	Publishes ten papers on good venues as the first author; low citations	Promising star
a_2	Co-authors ten papers with a_0 as the second author	Free-loader
a_3	Publish ten papers on good venues as the first author; cited by good papers	Prestigious expert; recognized by experts
a_4	Publish ten papers on good venues as the first author; cited by mediocre/poor papers	Expert; not recognized by experts
a_5	Publish ten papers on junk venues; cites his own papers a lot	Junk paper producer; spammer
a_6	Publish ten papers but none of them are related to the specific topic	Outsider

Table 5.6: A synthetic toy dataset of seven authors.

Constraints:

- a_0 and a_2 always co-author and a_0 is always the first author.
- a_1, a_3, a_4 and a_5 publish totally the same content (number of papers, phrases in each paper are the same). a_1 has very few citations, a_3's papers are cited a lot by good papers while a_4's papers are cited a lot by poor papers. Also, a_1, a_3 and a_4 publish on the same good venues while a_5 publish on the poor venues.

Effectiveness of *ExpertFinder*

To prove the effectiveness of *ExpertFinder* generative model, we mainly want to address two points: 1) Similar to the prevailing topic models like Latent Dirichlet Allocation (LDA), our model could effectively infer the ranking distributions of authors and venues for each subtopic. 2) Our model shows better performance when incorporating a topic hierarchy (pre-computed phrase ranking distribution).

Our toy experiment shows that if most of papers contain highly relevant phrases, incorporating topic hierarchy or not will not matter much since the phrase distribution for each subtopic could be inferred quite

precisely. However, if most papers only contain middle-level relevant phrases and some non-relevant phrases, which is closer to the real-world case, incorporating the topic hierarchy could play the role of "calibrator", which will improves the inference of author and venue ranking distribution.

Effectiveness of *BibRank*

To prove the *BibRank* ranking algorithm could actually work out the intuitions and heuristics we have discussed previously, we initialize the all authors with the same authority scores, which excludes the impact of *ExpertFinder* in the previous step. The result is shown in :

	a_0	a_1	a_2	a_3	a_4	a_5
before *BibRank*	0.166	0.166	0.166	0.166	0.166	0.166
after *BibRank*	0.209	0.173	0.106	0.228	0.183	0.010

Table 5.7: Authority scores of a_0 to a_5 before and after running *BibRank*.

The final rank (from high authority to low authority) is: $a_3 > a_0 > a_4 > a_1 >> a_2 > a_5$, which is Prestigious expert (recognized by other experts) > Prestigious expert > Expert (not recognized by other experts) > Promising star >> Free-loader > Junk paper producer (spammer). Specifically,

- a_5 publish totally the same content with a_1, a_3 and a_4, but has a much lower authority score, which shows *BibRank* effectively penalizes the authors who publish on junk venues and cite their own papers even if they seem productive.
- a_1 has not accumulate many citations but is ranked very close to the "prestigious experts" since his publishes on good venues. This shows that *BibRank* does not bury the young talents.
- a_0 and a_2 always collaborate on publishing papers but a_0 has a much higher authority score than a_2, which shows *BibRank* could effectively distinguish the efforts that co-authors contribute to the papers, hence filters out the free-loaders.
- a_3 and a_4 publish totally the same content but a_3 has a higher authority score than a_4, which is because a_3's papers are mostly cited by high quality papers while a_4's papers are mostly cited by mediocre or poor papers. This shows that *BibRank* actually differentiates the sources of citations, which follows the similar philosophy behind*PageRank*.

Chapter 6

Related Work

6.1 Topical Hierarchy Construction

Topical hierarchies, concept hierarchies, ontologies, etc., provide a hierarchical organization of data at different levels of granularity, and have many important applications. The related techniques can be broadly categorized as statistics-based or linguistic-based. Many studies are devoted to mining subsumption ('is-a') relationships[10]. Chuang and Chien[11] and Liu *et al.*[12] generate taxonomies of given keyword phrases by supplementing hierarchical clustering techniques with knowledge bases and search engine results. Wang *et al.*proposed CATHY, a statistics-based technique which constructs a topical hierarchy without resorting to external knowledge resources such as WordNet or Wikipedia. Later on, CATHYHIN[8] approach was developed that works with a heterogeneous information network and discovers multi-typed topical entities.

6.2 Topic Modeling

Considerable research has been conducted for investigating topic models or latent latent semantic structures for text mining. Hofmann[13] proposed the probabilistic latent semantic indexing (pLSI) and applies it to information retrieval (IR). Blei *et al.*[14] introduced a three-layer Bayesian network, called Latent Dirichlet Allocation (LDA). The basic generative process of LDA closely resembles pLSI except that in pLSI, the topic mixture is conditioned on each document while in LDA, the topic mixture is drawn from a conjugate Dirichlet prior that remains the same for all documents.

Some other work has been conducted for modeling both authorship and document contents simultaneously. In Author-Topic (AT) model[1], each author has a distribution over topics, unlike the simple topic model where each document has its own topic distribution. Each word is generated by selecting one of authors, sampling a topic from that authors topic distribution, and then sampling a word from that topics distribution over the vocabulary. McCallum *et al.* proposed Author-Recipient-Topic (ART) model for social network analysis based on LDA and AT models, adding key attribute that distribution over topics is con-

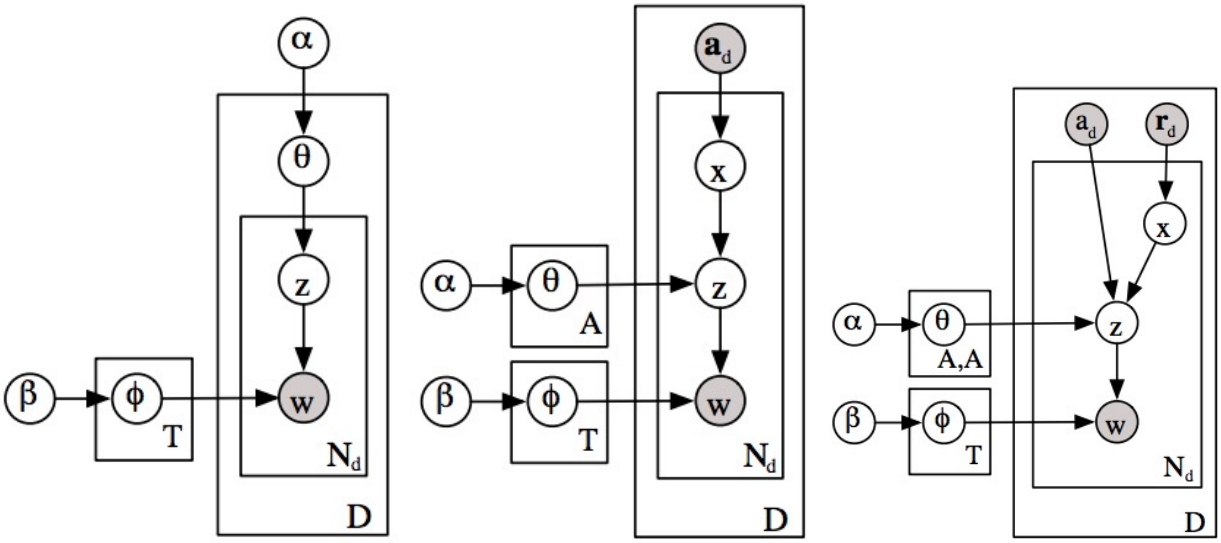
ditioned distinctively on both the content senders and receivers. In Author-Persona-Topic (APT) model[3], each author can write under one or more "personas", which are represented as independent distributions over hidden topics. Citation-Author-Topic (CAT) model[4] extends previous work by explicitly modeling the cited author information during the generative process. Tang *et al.* proposed Author-Conference-Topic (ACT) model and developed three strategies. In ACT1, each author is associated with a multinomial distribution over topics and each word in a paper and the conference stamp is generated from a sampled topic. In ACT2, each author-conference pair is associated with a multinomial distribution over topics and each word is then generated from a sampled topic. In ACT3, each author is associated with a topic distribution and the conference stamp is generated after topics have been sampled for all word tokens in a paper. The plate notations of these topic models are shown in 6.1.

6.3 Link Analysis

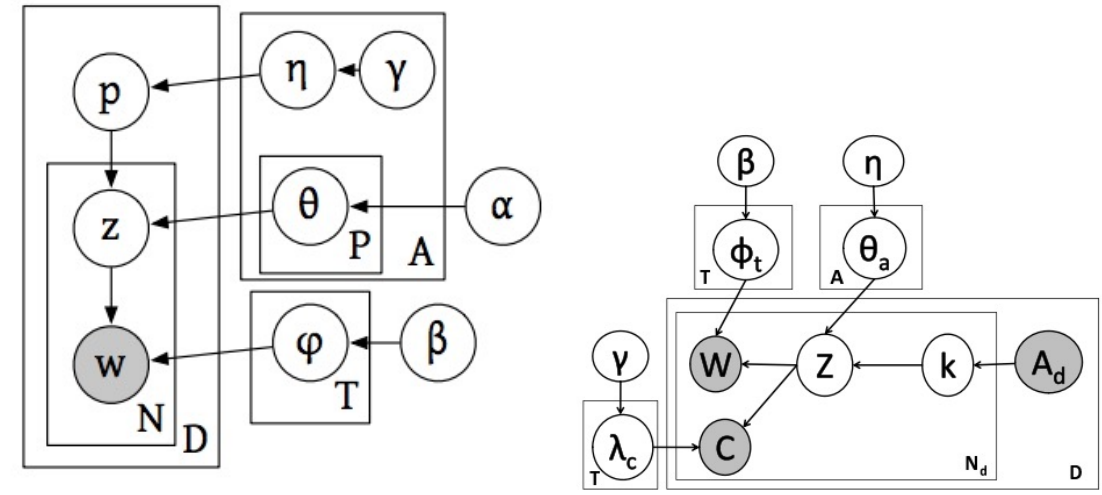
There are a variety of link analysis approaches aiming at evaluating the prestige of nodes in network structures, eg. World Wide Web, social network site, etc. PageRank algorithm, proposed by Page *et al.* [15] is one of the most famous. It provides a kind of peer assessment of the value of a Web page by taking into account not just the number of pages linking to it (in-degrees), but also the number of pages pointing to those pages, and so on. Thus, a link from a popular page is given a higher weighting than one from an unpopular page. Intuitively, the ranking in PageRank corresponds to the fraction of time a random walker would spend 'visiting' a page by iteratively following links from page to page.

A lot of work are conducted on the base of PageRank Algorithm. Topic-sensitive PageRank[16] computes a set of PageRank vectors, biased using a set of representative topics, to capture more accurately the notion of importance with respect to a particular topic. TwitterRank[17] extends Topic-sensitive PageRank by computing the transition probability as the similarity between two nodes instead of setting them uniformly. This idea fits into Twitter scenario very well because the more similar two Twitter users are, the more likely we will follow from one to the other. Nie *et al.* proposed PopRank, a domain-independent object-level link analysis model to rank the objects within a specific domain. This model captures the heterogeneous relationships between objects by specifically assigning a popularity propagation factor to each type of object relationship and study how different popularity propagation factors for these heterogeneous relationships could affect the popularity ranking.

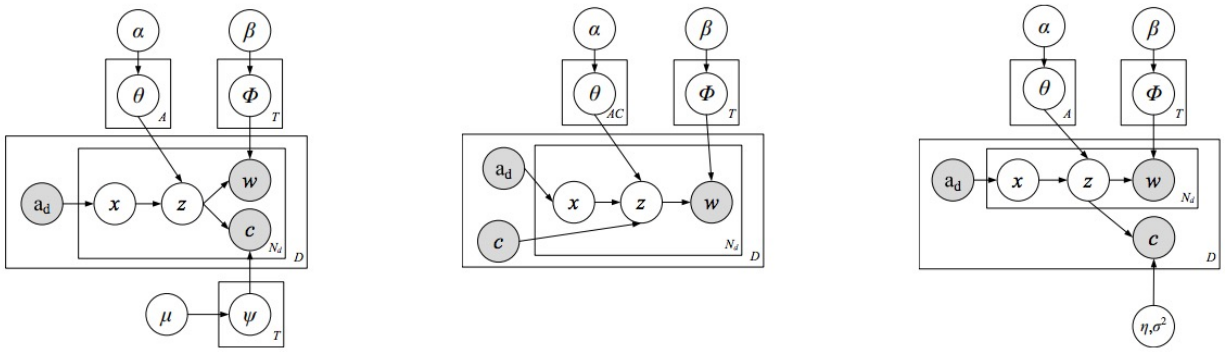
Another ranking algorithm similar to PageRank is HITS ("Hypertext induced topic selection")[18]. It also uses an iterative approach, but assigns two scores to each node: a *hub* score and an *authority* score. A



(a) Latent Dirichlet Allocation (LDA) (b) Author-Topicl (AT) (c) Author-Recipient-Topic (ART)



(d) Author-Persona-Topic (APT) (e) Citation-Author-Topic (CAT)



(f) Author-Conference-Topic (ACT)

Figure 6.1: Plate notations of topic models.

good hub is a node which links to many good authorities and a good authority is a node which is linked from many good hubs. The definition is recursive and converges after a few iterations.

Chapter 7

Conclusion

In this thesis, we address the problem of hierarchical expert finding in heterogeneous bibliographic network by constructing an expert hierarchy from given textual topical hierarchy. We develop a novel method *Hefbib* and provide details of two core components: the generative topic model *ExpertFinder* and the link analysis algorithm *BibRank*. Our approach reflect two basic properties of experts: relevancy and authority. We run our method on real-world DBLP dataset to evaluate its effectiveness and conduct case studies. We hope to refine the online query search module in the future work.

References

[1] Mark Steyvers, Padhraic Smyth, Michal Rosen-Zvi, and Thomas Griffiths. Probabilistic author-topic models for information discovery. In *KDD*, 2004.

[2] Andrew Mccallum, Andr Corrada-emmanuel, and Xuerui Wang. Topic and role discovery in social networks. In *IJCAI*, 2005.

[3] David Mimno and Andrew McCallum. Expertise modeling for matching papers with reviewers. In *KDD*, 2007.

[4] Yuancheng Tu, Nikhil Johri, Dan Roth, and Julia Hockenmaier. Citation author topic model in expert search. In *COLING*, 2010.

[5] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Extraction and mining of academic social networks. In *KDD*, 2008.

[6] Chi Wang, Marina Danilevsky, Nihit Desai, Yinan Zhang, Phuong Nguyen, Thrivikrama Taula, and Jiawei Han. A phrase mining framework for recursive construction of a topical hierarchy. In *KDD*, 2013.

[7] Jialu Liu, Jingbo Shang, Chi Wang, Xiang Ren, and Jiawei Ha. Mining quality phrases from massive text corpora. In *SIGMOD*, 2015.

[8] Chi Wang, Marina Danilevsky, Jialu Liu, Nihit Desai, Heng Ji, and Jiawei Han. Constructing topical hierarchies in heterogeneous information networks. In *ICDM*, 2013.

[9] Jonathan Chang, Sean Gerrish, Chong Wang, Jordan L Boyd-Graber, and David M Blei. Reading tea leaves: How humans interpret topic models. In *NIPS*, 2009.

[10] D. Lawrie and W. B. Croft. Discovering and comparing topic hierarchies. In *RIAO*, 2000.

[11] S.-L. Chuang and L.-F. Chien. A practical web-based approach to generating topic hierarchy for text segments. In *CIKM*, 2004.

[12] X. Liu, Y. Song, S. Liu, and H. Wang. Automatic taxonomy construction from keywords. In *KDD*, 2012.

[13] T.Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, 1999.

[14] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 2003.

[15] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: bringing order to the web. 1999.

[16] Taher H Haveliwala. Topic-sensitive pagerank. In *WWW*, 2002.

[17] Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He. Twitterrank: finding topic-sensitive influential twitterers. In *WSDM*, 2010.

[18] Jon M Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 1999.