*Social Network Extraction of Academic Researchers

Jie Tang, Duo Zhang, and Limin Yao
Department of Computer Science and Technology, Tsinghua University
{tangjie, zhangduo, ylm}@keg.cs.tsinghua.edu.cn

Abstract

This paper addresses the issue of extraction of an academic researcher social network. By researcher social network extraction, we are aimed at finding, extracting, and fusing the 'semantic'-based profiling information of a researcher from the Web. Previously, social network extraction was often undertaken separately in an ad-hoc fashion. This paper first gives a formalization of the entire problem. Specifically, it identifies the 'relevant documents' from the Web by a classifier. It then proposes a unified approach to perform the researcher profiling using Conditional Random Fields (CRF). It integrates publications from the existing bibliography datasets. In the integration, it proposes a constraints-based probabilistic model to name disambiguation. Experimental results on an online system show that the unified approach to researcher profiling significantly outperforms the baseline methods of using rule learning or classification. Experimental results also indicate that our method to name disambiguation performs better than the baseline method using unsupervised learning. The methods have been applied to expert finding. Experiments show that the accuracy of expert finding can be significantly improved by using the proposed methods.

1. Introduction

Social network services (SNSs) have been given much attention on the Web recently. As a typical online system, e.g., Facebook.com and MySpace.com, the user is required to enter a profile by her- or himself. The manual method can be used to construct a user-centered network, for example sending messages or focused communities such as music communities. Unfortunately, the method is not sufficient for mining in the Web 2.0 and Semantic Web. The information obtained solely from the user entered profile is sometimes incomplete or inconsistent. For example, users do not fill some information merely because they are not willing to fill the information.

Automatic extraction of the social network, for instance extraction of user profiles, is a promising solution to the problem, especially in some specific domains such as the researcher social network. This paper intends to conduct a thorough investigation on the issue of social network extraction of academic researchers. Specifically, it focuses on studying how to extract the profile for a researcher and how to disambiguate the researchers having the same name.

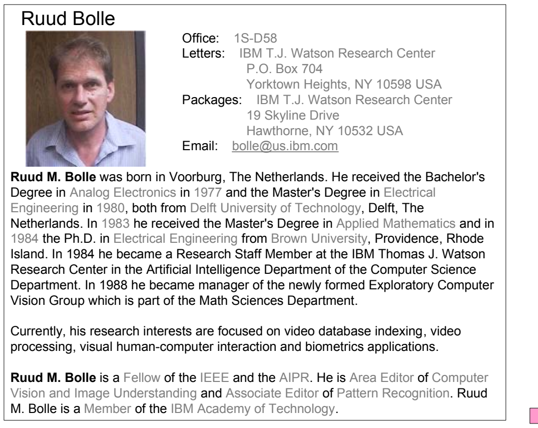
1.1. Motivating example

We begin by illustrating the problem with an example, drawn from an actual case of extracting researcher profiles in our developed system (http://www.arnetminer.org), which is also the initial motivation of the work. In this system, we intend to construct a 'semantic'-based social network for academic researchers. Specifically, we extract the basic information, contact information, and educational history of a researcher from the Web and create a researcher profile. We integrate the publication information into the profile from DBLP.

Construction of the researcher network by automatic extracting information from the Web can benefit many Web mining and social network applications. For example, if all the profiles are correctly extracted, we will have a large collection of *well-structured* data about real-world researchers. We can utilize the 'semantics'-based profiles to help enhance the mining such as expert finding.

Figure 1 shows an example of researcher profile extraction. The left part shows a researcher homepage which includes typical researcher profile information and a DBLP page which contains his published papers. The ideal extraction/integration results are shown in the right part of Figure 1.

* The work is supported by the National Natural Science Foundation of China under Grant No. 90604025.



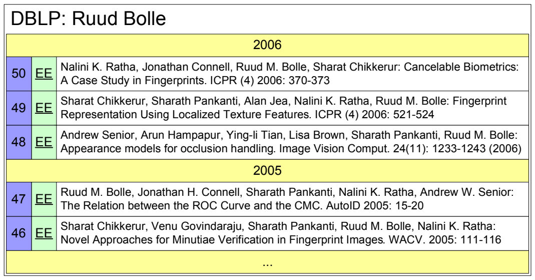
Ruud Bolle

Office: 1S-D58
Letters: IBM T.J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598 USA
Packages: IBM T.J. Watson Research Center
19 Skyline Drive
Hawthorne, NY 10532 USA
Email: bolle@us.ibm.com

Ruud M. Bolle was born in Voorburg, The Netherlands. He received the Bachelor's Degree in Analog Electronics in 1977 and the Master's Degree in Electrical Engineering in 1980, both from Delft University of Technology, Delft, The Netherlands. In 1983 he received the Master's Degree in Applied Mathematics and in 1984 the Ph.D. in Electrical Engineering from Brown University, Providence, Rhode Island. In 1984 he became a Research Staff Member at the IBM Thomas J. Watson Research Center in the Artificial Intelligence Department of the Computer Science Department. In 1988 he became manager of the newly formed Exploratory Computer Vision Group which is part of the Math Sciences Department.

Currently, his research interests are focused on video database indexing, video processing, visual human-computer interaction and biometrics applications.

Ruud M. Bolle is a Fellow of the IEEE and the AIPR. He is Area Editor of Computer Vision and Image Understanding and Associate Editor of Pattern Recognition. Ruud M. Bolle is a Member of the IBM Academy of Technology.



DBLP: Ruud Bolle

2006		
50	EE	Nalini K. Ratha, Jonathan Connell, Ruud M. Bolle, Sharat Chikkerur: Cancelable Biometrics: A Case Study in Fingerprints. ICPR (4) 2006: 370-373
49	EE	Sharat Chikkerur, Sharath Pankanti, Alan Jea, Nalini K. Ratha, Ruud M. Bolle: Fingerprint Representation Using Localized Texture Features. ICPR (4) 2006: 521-524
48	EE	Andrew Senior, Arun Hampapur, Ying-li Tian, Lisa Brown, Sharath Pankanti, Ruud M. Bolle: Appearance models for occlusion handling. Image Vision Comput. 24(11): 1233-1243 (2006)
2005		
47	EE	Ruud M. Bolle, Jonathan H. Connell, Sharath Pankanti, Nalini K. Ratha, Andrew W. Senior: The Relation between the ROC Curve and the CMC. AutoID 2005: 15-20
46	EE	Sharat Chikkerur, Venu Govindaraju, Sharath Pankanti, Ruud M. Bolle, Nalini K. Ratha: Novel Approaches for Minutiae Verification in Fingerprint Images. WACV. 2005: 111-116
...		

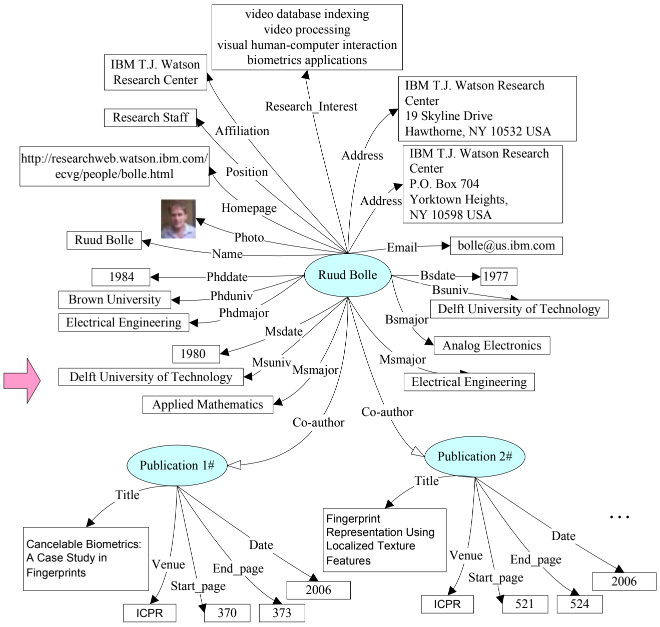


Figure 1. An example of researcher profile extraction

In this paper, we target at dealing with two challenges: (1) How to extract the profile information from the Web and (2) how to integrate the profile information extracted from different sources.

For extraction of the profile information, the manual entering mean for each researcher is obviously tedious and time consuming. Recent work has shown the feasibility and promise of information extraction technologies for extracting the structured data from the Web, and it is possible to use the methods to extract the profile of a researcher. However, most of the existing methods employed a predefined rule or a specific machine learning model to identify each type of information independently. It is *highly ineffective* to use the separated methods to do researcher profile extraction due to the natural disadvantages of the method: (1) For each property in the profile, one has to define a specific rule or supervised learning model. Therefore, there may be many different rules/models, which are difficult to maintain; (2) The separated rules/models cannot take advantage of dependencies across different properties. The properties are often dependent with each other. For instance, in Figure 1 identifying the text 'Electrical Engineering' as *Msmajor* will greatly increase the probability of the text 'Delft University of Technology' to be identified as *Msuniv*. Consequently, how to *effectively* identify the profile information from the Web becomes a challenging issue.

For integration of the profile information from different sources, we focus on the name disambiguation problem. Existing methods include heuristic rules, classification-based supervised method, and clustering-based unsupervised method. However, it is also *not effective* to directly employ the existing methods in researchers' profile integration. This is because: (1) The heuristic rule based method requires the user to define a specific rule for each specific type of ambiguity problem, which is not adaptive for different situations; (2) The supervised method trains a user-dependent model for a certain person and thus cannot be adapted to the other person; and (3) The clustering based unsupervised method can deal with different persons simultaneously, however, it cannot make use of the supervised information.

1.2. Our solution

In this paper, we aim to conduct a thorough investigation on the problem. First, we formalize researcher network extraction as a process of identifying relevant Web pages, extracting profile information, and fusing the profile information from different sources. Secondly, we employed a classifier to identify the relevant Web pages. Then we propose a unified approach to extract the profile information from the identified Web pages on the basis of tagging. Specifically, we view the problem as that of assigning tags to the input texts, with a tag representing one profile property. Furthermore, we propose a constraint-based probabilistic model to name disambiguation. The model can incorporate any types of domain background knowledge or supervised information (e.g., user feedbacks) as constraints to improve the performances of disambiguation. We define six types of constraints. To the best of our knowledge, research profiling in a unified approach and name disambiguation using a constraint-based probabilistic model have not been investigated previously.

Experimental results indicate that our method significantly outperforms the methods of using separated models for profile extraction. Experimental results also indicate that our disambiguation method can significantly outperform the unsupervised method. We applied our methods to expert finding. Experimental results show that our methods of profile extraction and name disambiguation can indeed enhance expert finding (+22% in terms of MAP).

Our contributions in this paper include: (1) formalization of the problem of researcher network extraction, (2) proposal of a unified tagging approach to researcher profiling, (3) proposal of a constraint-based probabilistic model to name disambiguation, and (4) empirical verification of the effectiveness of the proposed approaches.

The rest of the paper is organized as follows. In Section 2, we formalize the extraction problem. In Section 3, we explain our approaches and in Section 4 we give the experiments. Before concluding the paper in Section 6, we introduce related work.

2. Researcher social network extraction

We use the data from the ArnetMiner system for study. The system aims at providing a social networking platform for academic researchers. It has gathered 448,289 researchers. Our statistical study on the half million researchers shows that about 70.60% of the researchers have at least one homepage or a Web page that introduces them, which implies that extraction of the profile from the Web is *feasible*. For the ambiguity problem, we have examined 30 random person names and found that more than 60% of the names have the ambiguity problem.

We define the schema of the researcher profile (as shown in Figure 2), by extending the FOAF ontology [5]. (Cf. Figure 1 for sample instances.) In the profile, two classes, 24 properties and two relations are defined.

It is non-trivial to perform the researcher network extraction from the Web. We here describe the two key issues we are going to deal with: researcher profile extraction and name disambiguation.

(1) Researcher profile extraction. We produced statistics on randomly selected 1K researchers. We observed that 85.6% of the researchers are faculties of universities and 14.4% are from company research centers. For researchers from the same company, they often have a template-based homepage. However, different companies have absolutely different templates. For researchers from universities, the layout and the content of the homepages vary largely depending on the authors. We have also found that 71.9% of the 1K Web pages are researchers' homepages and the rest are pages introducing the researchers. Characteristics of the two types of pages significantly differ from each other.

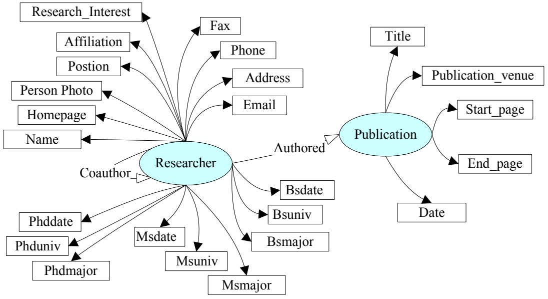


Figure 2. The schema of the researcher profile

We analyzed the content of the Web pages and found that about 40% of the profile properties are presented in tables or lists and the others are presented in natural language. This also means a method without using the global context information in the page would be ineffective. Statistical study also unveils that (strong) dependencies exist between different profile properties. For example, there are 1325 cases (14.5%) in our data that the property label of the tokens need use the extraction results of the other tokens. An ideal method should consider processing all the subtasks together.

(2) Name disambiguation. We do not perform extraction of publications directly from the Web. Instead, we integrate the publication data from existing online data source. We chose DBLP bibliography (dblp.uni-trier.de/), which is one of the best formatted and organized bibliography datasets. DBLP covers approximately 800,000 papers from major Computer Science publication venues. In DBLP, authors are identified by their names. For integrating the researcher profiles and the publications data, we use researcher names and the author names as the identifier. The method inevitably has the ambiguity problem (different researchers have the same name).

We here give a formal definition of the name disambiguation task in our context. Given a person name a, we denote all publications containing the author named a as $P=\{p_1, p_2, \ldots, p_n\}$. For each publication p_i, it has six attributes as shown in Table 1.

Here, each name $a_i^{(j)}$ has an affiliation $a_i^{(j)}$.*affiliation* and an email $x_i^{(j)}$.*email*. We call the first author name

$a_i^{(0)}$ as the *principal author* and the others *secondary authors*. Suppose there existing k actual researchers $\{y_1, y_2, \ldots, y_k\}$ having the name a, our task is to assign these n publications to their real researcher y_i.

Table 1. Attributes of each publication

Attribute	Description
p_i.*title*	The title of p_i
p_i.*conference*	The published conference/journal of p_i
p_i.*year*	The year when p_i is published
p_i.*abstract*	The abstract of p_i
p_i.*authors*	The authors name set of p_i $\{a_i^{(0)}, a_i^{(1)}, \ldots a_i^{(u)}\}$
pi.references	The reference set of p_i which is denoted as REF_i

Next, we define a rule set $R=\{r_1, r_2, \ldots, r_m\}$ and correspondingly a constraint set $C=\{c_1, c_2, \ldots, c_m\}$. We say a pair of publication p_i and p_j satisfies a constraint c_l if they satisfy the rule r_l, i.e.

$$c_l(p_i, p_j) = \begin{cases} 1 & \text{if } p_i \text{ and } p_j \text{ satisfy rule } r_l \\ 0 & \text{otherwise} \end{cases}$$

We extracted the attribute values of each paper from several digital libraries, e.g., IEEE, Springer, and ACM. We used heuristics to perform the extraction.

3. Our approach

There are three steps in our approach: relevant page identification, researcher profiling, and publication integration. In relevant page identification, given a researcher name, we first get a list of web pages by a search engine (we used Google API) and then identify the homepage/introducing page using a classifier.

In researcher profile extraction, we propose a unified approach. The approach can incorporate dependencies between different types of profile properties to do better extraction.

In publication integration, we propose a constraint-based probabilistic model to name disambiguation.

The first issue has been intensively studied as explained in Section 5. The latter two issues have not been thorough investigated previously and are the main focus of our work. Both of the two proposed approaches to researcher profile extraction and name disambiguation are based on the theory of Markov Random Field.

3.1. Markov random field

Markov Random Field (MRF) is a probability distribution of labels (hidden variables) that obeys the Markov property. It can be formally defined as follows. **MRF Definition.** *Let $G = (V, E)$ be a graph such that $Y=(Y_v)_{v\in V}$, so that Y is indexed by the vertices of G. Then (X, Y) is a Markov random field in case, when the random variable Y_v obeys the Markov property with respect to the graph: $p(Y_v|Y_w, w\neq v) = p(Y_v|Y_w, w\backsim v)$, where $w\backsim v$ means that w and v are neighbors in G.*

Many special cases of MRF can be developed, for example, Conditional Random Fields (CRFs) [17] and Hidden Markov Random Fields (HMRF) [3].

3.2. A unified approach to profiling

3.2.1. Process. The approach consists of two steps: preprocessing and tagging. In preprocessing, (A) we separate the text into tokens and (B) we assign possible tags to each token. The tokens form the basic units and the pages form the sequences of units in the tagging problem. In tagging, given a sequence of units, we determine the most likely corresponding sequence of tags by using a trained tagging model. (The type of the tags corresponds to the property defined in Figure 2.) In this paper, as the tagging model, we make use of Conditional Random Fields (CRFs). Next we describe the steps (A) and (B) in detail.

(A). We identify tokens in the Web page by using heuristics. We define five types of tokens: 'standard word', 'special word', '<image>' token, term, and punctuation mark. Standard words are unigram words in natural language. Special words [21] include email, URL, date, number, percentage, words containing special symbols (e.g. 'Ph.D.' and '.NET'), unnecessary tokens (e.g. '===' and '###'), etc. We identify special words by using regular expressions. '<image>' tokens are '<image>' tags in the HTML file. We identify it by parsing the HTML file. Terms are base noun phrases extracted from the Web pages. We developed a tool based on technologies proposed in [26].

(B). We assign possible tags to each token based on the token type. For example, for standard word, we assign all possible tags (each tag represents a property). For special word, we assign tags: *Position*, *Affiliation*, *Email*, *Address*, *Phone*, *Fax*, and *Bsdate*, *Msdate*, and *Phddate*. For '<image>' token, we assign two tags: *Photo* and *Email* (an email is likely to be shown as an image).

In this way, each token can be assigned several possible tags. Using the tags, we can perform most of the profiling processing (conducting 16 subtasks defined in Figure 2). We do not conduct research interest extraction using the proposed approach, although we could do it in principle. There are two reasons: first, we observed only one fifth (21.3%) of researchers provide the research interest on homepages; secondly, research interest is usually implied by the other profile properties, e.g., papers published by the researcher or research projects he/she is involved in.

3.2.2. CRF model. We employ Conditional Random Fields (CRF) as the tagging model. CRF is a special case of MRF. CRF is a conditional probability of a sequence of labels y given a sequence of observations tokens [18]. In tagging, the CRF model is used to find the sequence of tags Y^* having the highest likelihood $Y^* = \max_Y P(Y|X)$, with the Viterbi algorithm.

In training, the CRF model is built with labeled data and by means of an iterative algorithm based on Maximum Likelihood Estimation.

3.2.3. Features. Three types of features were defined: content features, pattern features, and term features.

1. Content features

For a standard word, the content features include:

(1) Word features. The features represent whether the current token contains a word or not.

(2) Morphological features. The features represent morphologies of the token, e.g. whether the token is capitalized.

For a '<image>' token in the HTML file, the content features include:

(1) Image size feature. It indicates the size of the image.

(2) Image height/width ratio feature. It represents the ratio of the height to the width of the current image.

(3) Image format feature. It indicates the format of the image (e.g. 'JPG', 'BMP').

(4) Image color feature. It represents the number of the 'unique color' used in the image and the number of bits used for per pixel (e.g. 32, 24, 16, 8, and 1).

(5) Face feature. It represents whether the current image contains a person face. We use a tool from http://opencvlibrary.sf.net to detect the face in a picture.

(6) Image filename feature. It represents whether the image filename contains the research name.

(7) Image 'alt' feature. It represents if the 'alt' attribute of the '<image>' token contains the research name.

(8) Image positive word features. The features indicate whether the image filename contains positive keywords like 'myself' and 'biography'.

(9) Image negative word features. The features indicate whether the image filename contains negative keywords like 'logo', 'banner', and 'ads'.

2. Pattern features

Pattern features are defined for each token.

(1) Positive word features. The features represent if the current token contains positive fax keywords like 'Fax:', positive position keywords like 'Manager'.

(2) Special token feature. It represents whether the current token is a special word.

(3) Researcher name feature. The feature represents if the current token contains the researcher name.

3. Term features

Term features are defined only for term token.

(1) Term features. The features represent whether the term contains a base noun phrase or not.

(2) Dictionary features. The features represent whether the term contains a word in a dictionary.

We can easily incorporate the features defined above into our model by defining Boolean-valued feature functions. In total, 108,409 features were used in our experiments.

3.3. A constraint-based probabilistic model to name disambiguation

3.3.1. Process. Our method is based on a probabilistic model using Hidden Markov Random Fields (HMRF). This model incorporates constraints and a parameterized-distance measure. The disambiguation problem is cast as assigning a tag to each paper with each tag representing an actual researcher y_i.

Specifically, we define the a-posteriori probability as the objective function. We aim at optimizing the objective function. We incorporate six types of constraints into the objective function. If one paper's label assignment violates a constraint, it will be penalized in some sense, which in turn affects the disambiguation result.

3.3.2. Formalization using HMRF. A HMRF based semi-supervised framework is first introduced by [3]. HMRF is a generative model, which describes the joint probabilities. Based on Bayesian rule, the posterior probability of researcher labels Y can be written as:

$$P(Y \mid X) \propto P(Y)P(X \mid Y)$$

Again, by the Hammersley-Clifford theorem [14], $P(Y)$ can be expressed as:

$$P(Y) = \frac{1}{Z_1}\exp(-\sum\nolimits_{i,j} w_k c_k(y_i, y_j))$$

where $c_k(y_i, y_j)$ denotes a constraint of x_i and x_j; w_k is the parameter; Z_1 is the normalization factor.

For simplification, $P(X|Y)$ can be restricted as the exponential form [3]:

$$P(X \mid Y) = \frac{1}{Z_2}\exp(-\sum\nolimits_i D(x_i, y_i))$$

where $D(x_i, y_i)$ is the distance between the paper x_i and its assigned researcher y_i; Z_2 is the normalization factor.

Putting $P(Y)$ and $P(X|Y)$ together, we can obtain

$$P(Y \mid X) = \frac{1}{Z}\exp\left(\sum_i D(x_i, y_i) + \sum_{i,j,c_k \in C} w_k c_k(y_i, y_j)\right)$$

where Z=Z_1Z_2.

The key issue here is how to define *constraints* for effectively performing the disambiguation task.

3.3.3. Constraint selection. We define six types of constraints based on the characteristic of publication dataset. Table 2 shows the constraints.

All these constraints are defined between two papers p_i and p_j. The first constraint c_1 means the principal authors of two papers are from the same organization. Constraint c_2 means two publications have a secondary author with the same name, and the constraint c_3 means whether a paper cites another paper. Constraint c_4 means whether principal authors of the two publications have the same email address (this is a stronger constraint than the others). Constraint c_5 denotes user interaction.

Table 2. Constraints used in our approach

C	***W***	**Constraint Name**	**Description**
c_1	w_1	CoOrg	$a_i^{(0)}.affiliation = a_j^{(0)}.affiliation$
c_2	w_2	CoAuthor	$\exists\, r, s>0, a_i^{(r)}=a_j^{(s)}$
c_3	w_3	Citation	p_i cites p_j or p_j cites p_i
c_4	w_4	CoEmail	$a_i^{(0)}.email = a_j^{(0)}.email$
c_5	w_5	Feedback	Constraints from user feedback
c_6	w_6	τ-CoAuthor	one common author in τ extension

We use an example to explain constraint c_6. Suppose p_i has authors 'David Mitchell' and 'Andrew Mark', and p_j has authors 'David Mitchell' and 'Fernando Mulford'. If 'Andrew Mark' and 'Fernando Mulford' also coauthor one publication, then we say p_i and p_j have a 2-CoAuthor constraint. We construct a matrix M (as shown in Figure 3) to test whether two papers have a τ-CoAuthor constraint.

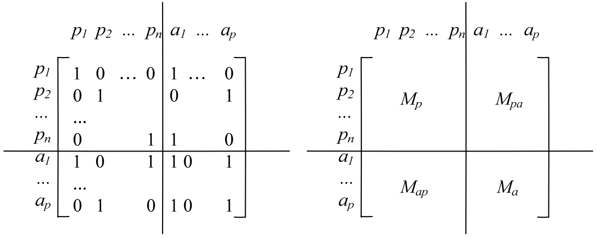


Figure 3. Matrix *M* for c_6 constraint

In matrix M, p_1,p_2,…,p_n are publications with an author named a. a_1,a_2,…,a_p is the union set of all p_i.*authors*, i=1,2,…n, i.e.

$$\{a_1, a_2, ..., a_p\} = \bigcup_{i=1}^{n} p_i.authors = \bigcup_{i=1}^{n} \{a_i^{(1)}, a_i^{(2)}, ..., a_i^{(u_i)}\}$$

Note that a_1,a_2,…,a_p does not include $a_i^{(0)}$. The sub matrix M_p indicates the relationship between p_1,p_2,…,p_n and initially it is an identity matrix. In sub matrix M_{pa}, an element on row p_i and column a_j is equal to 1 if and only if $x_j \in p_i.authors$, otherwise 0. The matrix M_{ap} is symmetric to M_{pa}. Sub matrix M_a indicates the coauthorship among a_1,a_2,…,a_p. The element on row x_i and column x_j is equal to 1 if and only if a_i and a_j coauthor one publication in our database (not just limited to p_1,p_2,…,p_n), otherwise 0.

By multiplying M by itself, i.e. $M^{(1)}$=M•M, the element on row p_i and column p_j becomes 1 if they have at least one common secondary author. Thus, M shows 1-CoAuthor constraints between papers. Similarly, $M^{(2)}$=$M^{(1)}$•M indicates 2-CoAuthor constraints between papers. Likewise for the τ-CoAuthor constraints. If p_i and p_j have both τ_1-CoAuthor and τ_2-CoAuthor (τ_1<τ_2) constraint, we only consider the τ_1-CoAuthor constraint.

Next, we set the weight for each type of constraint empirically. For example, we assign c_2 constraint Co-Author a relatively high weight and assign w_6 as the τ power of w_2, i.e. w_6= w_2^{τ}. Emails can be regarded as unique identifiers for people, so we assign w_4 the largest value. User's feedback is another strong constraint. The larger the weight, the greater the impact of that constraint is. In our experiment, we set w_1 ~ w_6 as 0.5, 0.7, 0.6, 1.0, 0.9, 0.7^{τ} respectively.

3.3.4. EM framework. Three tasks are executed by the Expectation Maximization method: learning parameters of the distance measure, re-assignment of paper to researchers, and the update of researcher representatives y_h.

We define our distance function $D(\boldsymbol{x}_i, \boldsymbol{x}_j)$ as follows:

$$D(\boldsymbol{x}_i, \boldsymbol{x}_j) = 1 - \frac{\boldsymbol{x}_i^T \mathbf{A} \boldsymbol{x}_j}{\| \boldsymbol{x}_i \|_{\mathbf{A}} \| \boldsymbol{x}_j \|_{\mathbf{A}}}, \text{ where } \| \boldsymbol{x}_i \|_{\mathbf{A}} = \sqrt{x_i^T \mathbf{A} \boldsymbol{x}_j}$$

here **A** is a parameter matrix. For simplification, we define it as a diagonal matrix.

The EM process can be summarized as follows: in the E-step, given the current researcher representatives, every paper is assigned to the researcher by maximize $p(Y|X)$. In the M-step, the researcher representative y_h is re-estimated from the assignments to maximize $p(Y|X)$ again, and the distance measure is updated to maximize $P(Y|X)$.

In the initialization of our EM framework, we first cluster publications into disjoint groups based on the constraints over them, i.e. if two publications have a constraint, then they are assigned to the same researcher. Therefore, we first get λ groups. If λ is equal to our actual researcher number k, then these λ groups are used as our initial assignment. If λ < k, we choose (k - λ) random assignment. If λ > k, we cluster the nearest group until there are only k groups left.

In the E-step, assignments of data points to researchers are updated to maximize the $p(Y|X)$. A greedy algorithm is used to sequentially update the assignment for each paper. The algorithm performs assignments in random order for all papers. Each paper x_i is assigned to y_h that maximize the function:

$$f(y_h, x_i) = D(x_i, y_h) + \sum_{j, c_k \in C} [w_k c_k (y_i, y_j)]$$

The assignment of a paper is performed while keeping assignments of the other papers fixed. The assignment process is repeated after all papers are assigned. This process runs until no paper changes its assignment between two successive iterations.

In the M-step, each researcher representative is updated by the arithmetic mean of its points:

$$y_h = \frac{\sum_{i:y_i=h} \boldsymbol{x}_i}{\| \sum_{i:y_i=h} \boldsymbol{x}_i \|_{\mathbf{A}}}$$

Then, each parameter a_{mm} in **A** is updated by (only parameters on the diagonal) $a_{mm} = a_{mm} + \eta \dfrac{\partial f(y_h, x_i)}{\partial a_{mm}}$.

4. Experimental results

4.1. Experimental setting

4.1.1. Data sets. For profiling experimentation, we randomly chose in total 1K researcher names from our researcher network system. We used the method described in Section 3 to find the researchers' homepages or introducing pages. The F1-score of the process is 92.39%. If the method cannot find a Web page for a researcher, we remove the researcher name from the data set. We finally obtained 898 Web pages.

Seven human annotators conducted annotation on the Web pages. A spec was created to guide the annotation process. For disagreements in the annotation, we conducted 'majority voting'.

We produced statistics on the data set. In summary, 86.41% of the Web pages contain at least five properties and 96.44% contain four. We omit the details due to space limitation.

For name disambiguation, we created two datasets from our database, namely Abbreviated Name dataset and Real Name dataset. The first dataset was collected by querying 10 abbreviated names in our database. All the abbreviated names are created by simplifying the original names to its first name initial and last name, for example, 'Cheng Chang' to 'C. Chang'. The simplifying form is popular in bibliographic records. Statistics of this dataset is shown in Table 3.

Another dataset is constructed by querying two person names 'Jing Zhang' and 'Yi Li'. The purpose of constructing the small dataset is to analyze contributions of the six types of constraints we defined. 'Jing Zhang' has totally 54 publications by 25 different researchers and 'Yi Li' has 42 publications by 22 different researchers.

Table 3. Abbreviate Name dataset

Abbr. Name	#Public-ations	#Actual Researcher	Abbr. Name	#Public-ations	#Actual Researcher
C. Chang	402	97	M. Hong	108	30
G. Wu	152	46	X. Xie	136	36
K. Zhang	293	40	P. Xu	39	5
J. Li	551	102	H. Xu	182	60
B. Liang	55	14	W. Yang	263	82

4.1.2. Evaluation measures. In the experiments, we conducted evaluations in terms of precision, recall, and F1-measure (for definitions, see for example [25]). By comparison of the other work, we also give statistical significance estimates using Sign Test [13].

4.1.3. Implementation of baseline methods. We defined baselines for researcher profile extraction and name disambiguation.

For researcher profile extraction, we used the rule learning based approach and the classification based approach as baselines. For the former approach, we employed the Amilcare system [6]. The system is based on a rule induction algorithm, called LP^2. For the latter approach, we trained a classifier for identifying the values of each property. We employed Support Vector Machines (SVM) [9] as the classification model. We used the same features as those in our unified model.

Both of the two baselines perform extraction of each profile property independently. When there is a conflict between the outcomes of two classifiers, we adopt the result with higher predicting score.

To test how dependencies between different types of properties affect the performance of profiling, we also conducted experiments using the unified model by removing the transition features (Unified_NT).

For name disambiguation, we defined a baseline based on previous work [22] (except that [22] also uses a search engine to help the disambiguation). The baseline uses a hierarchical clustering algorithm to group the papers together. Then we view the grouped papers as the disambiguation results. We suppose that the number of persons k is provided empirically.

4.2. Researcher profile experiments

4.2.1. Results. Table 4 shows the five-fold cross-validation results. Our method outperforms the baseline method. We can also see that the performance of the unified method decreases when removing the transition features (Unified_NT).

Table 4. Performances of researcher profiling (%)

Profiling Task	Unified			Unified_NT			SVM			Amilcare		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Photo	90.32	88.09	89.11	89.22	88.19	88.64	87.99	89.98	88.86	97.44	52.05	67.86
Position	77.53	63.01	69.44	73.99	57.67	64.70	78.62	55.12	64.68	37.50	61.71	46.65
Affiliation	84.21	82.97	83.52	74.09	70.42	72.16	78.24	70.04	73.86	42.68	81.38	55.99
Phone	89.78	92.58	91.10	74.86	83.08	78.72	77.91	81.67	79.71	55.79	72.63	63.11
Fax	92.51	89.35	90.83	73.03	57.49	64.28	77.18	54.99	64.17	84.62	79.28	81.86
Email	81.21	82.22	80.35	81.66	70.32	75.47	93.14	69.18	79.37	51.82	72.32	60.38
Address	87.94	84.86	86.34	77.66	72.88	75.15	86.29	69.62	77.04	55.68	76.96	64.62
Bsuniv	74.44	62.94	67.38	64.08	53.16	57.56	86.06	46.26	59.54	21.43	20.00	20.69
Bsmajor	73.20	58.83	64.20	67.78	53.68	59.18	85.57	47.99	60.75	53.85	18.42	27.45
Bsdate	62.26	47.31	53.49	50.77	34.58	40.59	68.64	18.23	28.49	17.95	16.67	17.28
Msuniv	66.51	51.78	57.55	59.81	40.06	47.49	89.38	34.77	49.78	15.00	8.82	11.11
Msmajor	69.29	59.03	63.35	69.91	56.56	61.92	86.47	49.21	62.10	45.45	20.00	27.78
Msdate	57.88	43.13	48.96	48.11	36.82	41.27	68.99	19.45	30.07	30.77	25.00	27.59
Phduniv	71.22	58.27	63.73	60.19	48.23	53.11	82.41	43.82	57.01	23.40	14.29	17.74
Phdmajor	77.55	62.47	67.92	71.13	51.52	59.30	91.97	44.29	59.67	68.57	42.11	52.17
Phddate	67.92	51.17	57.75	50.53	36.91	42.49	73.65	29.06	41.44	39.13	15.79	22.50
Overall	**84.98**	**81.90**	**83.37**	75.04	69.41	72.09	81.66	66.97	73.57	48.60	59.36	53.44

We conducted sign tests on the extraction result for each property, which indicate that all the improvements of Unified over Amilcare, SVM, and Unified_NT are statistically significant (*p* << 0.01).

4.2.2. Contribution of features. We investigated the contribution of each feature type in profile extraction. We employed only content features, content+term features, content+pattern features, and all features to train the models and conducted the profile extraction.

Figure 4 shows the average F1-score of profile extraction with different feature types. We see that solely using one type of features alone cannot accomplish accurate profile extraction. The results also unveil the reason of the high performance in the extraction achieved by our method.

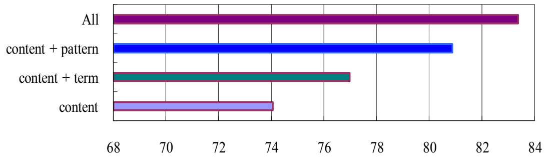


Figure 4. Contribution of features

4.2.3. Discussion. Our method outperforms Amilcare and SVM in most of the subtasks, especially in the subtasks that have strong dependencies with each other.

The baseline methods suffered from ignorance of the dependencies between the subtasks. For example, there were 337 cases (25.41%) in which *Address* identification needs to use the results of *Affiliation*. However, the baselines cannot make use of the dependencies, as it conducts all the subtasks independently. Our method benefits from the ability of modeling dependencies between subtasks. Table 4 shows that by leveraging the dependencies, our method outperforms the method without using (Unified_NT) by 10.28% in terms of F1-score.

Although we conducted error analysis on the results, we omit the details here due to space limitation and will report them in an expanded version.

4.3. Name disambiguation experiments

4.3.1. Results. The performances of our method and the baseline method on the Abbreviation Name dataset are shown in Table 5.

Table 5. Results on Abbreviate Name Dataset

Name	Baseline			Our Approach		
	Prec.	Rec.	F1	Prec.	Rec.	F1
C. Chang	0.65	0.59	0.62	0.73	0.67	0.70
G. Wu	0.71	0.62	0.66	0.75	0.75	0.75
K. Zhang	0.75	0.60	0.67	0.79	0.71	0.75
J. Li	0.62	0.52	0.57	0.66	0.59	0.62
B. Liang	0.82	0.76	0.79	0.85	0.89	0.87
M. Hong	0.79	0.65	0.71	0.82	0.75	0.78
X. Xie	0.77	0.73	0.75	0.83	0.82	0.82
P. Xu	0.89	0.95	0.92	0.94	1.00	0.97
H. Xu	0.65	0.59	0.62	0.73	0.67	0.70
W. Yang	0.71	0.62	0.66	0.75	0.75	0.75
Avg.	0.75	0.60	0.67	**0.79**	**0.71**	**0.75**

The proposed method outperforms the baseline method by 8.0% in terms of F1-measure.

4.3.2. Contribution of constraints. We investigated the contribution of each type of constraints in name disambiguation. Figure 5 shows the F1-score of 'Jing Zhang' and 'Yi Li' on the Real Name dataset with various combinations of constraints. We can see that the CoAuthor constraint contributes a lot to the results. It can be also seen that all the constraints we defined can enhance the final performance.

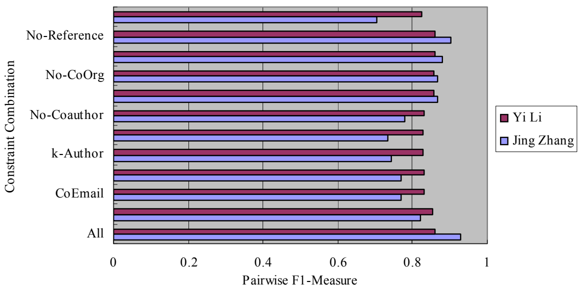


Figure 5. Contribution of constraints

4.4. Expert finding experiments

To further evaluate the effectiveness of our method, we applied it to expert finding. The task of expert finding is to identify persons with some given expertise or experience. In this task, we intend to test if the extracted profiles and the disambiguation results can be used to enhance expert finding.

We evaluated expert finding results without profile extraction (RPE) and disambiguation (ND) and the results by adding them one by one (+RPE and +ND). We selected 12 topics for finding experts from the system. We conducted evaluation in terms of P@5, P@10, P@20, P@30, *R*-prec, mean average precision (*MAP*), *bpref*, and mean reciprocal rank (*MRR*) [10].

Figure 6 shows the results of expert finding. We see that significant improvements can be obtained by using our methods. For example, in terms of mean average precision (MAP), 20% improvements can be obtained using profile extraction results (+RPE). With our name disambiguation method (+ND), MAP can be again improved by 2%.

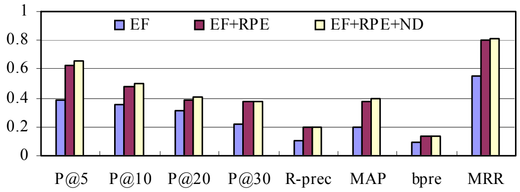


Figure 6. Performances of expert finding

5. Related work

5.1. Person profile extraction

Several research efforts have been made for extracting profile information of a person. For example, Yu et al. propose a cascaded information extraction framework for identifying personal information from resumes [27]. In their approach, a resume is first segmented into consecutive blocks attached with labels indicating the information type. And then, the detailed information such as *Address* and *Email*, are identified in certain blocks. The Artequakt system [1] employed a rule based extraction system called GATE [11] to extract entity and relation information from the Web. However, most of the previous works view the profile extraction as several separate issues and conduct a more or less ad-hoc manner. To the best of our knowledge, no previous work has been done on researcher profiling using a unified approach.

Considerable efforts have been placed on extraction of contact information from emails or the Web. For example, Kristjansson et al. developed an interactive information extraction system to assist the user to populate a contact database from emails [17]. Tang et al. propose a cascaded method for detecting signatures from emails [23]. See also [2]. Contact information extraction is a subtask of profile extraction, thus it significantly differs from profile extraction.

Several systems have been developed for searching for papers, for example, scholar.google libra.msra, citeseer.ist.psu, and dblife.cs.wisc. However, all the systems are focusing on providing services for searching publications rather than person.

Many information extraction methods have been proposed. Hidden Markov Model (HMM) [12], Maximum Entropy Markov Model (MEMM) [19], Conditional Random Field (CRF) [18], Support Vector Machines (SVM) [9], and Voted Perceptron [8] are widely used models. See [24] for an overview.

5.2. Name disambiguation

A number of approaches have been proposed to name disambiguation in different domains.

For example, [4] tries to distinguish web pages to different persons with the same name. They present two unsupervised frameworks for solving this problem: one is based on link structure of the Web pages and the other uses Agglomerative/Conglomerative double clustering method. See also [20]. The methods are based on unsupervised clustering. They cannot incorporate all types of constraints.

There are also many works focusing on name disambiguation in publication data. For example, Han et al. propose an unsupervised learning approach using K-way spectral clustering method [16]. They calculate a Gram matrix for each name dataset and apply K way spectral clustering algorithm to the Gram matrix to get the result. See also [22]. The type of method uses a parameter-fixed distance metric in their clustering algorithm, while parameters of our distance metric can be learned during the disambiguation process.

Two supervised methods are proposed in [15] based on Naïve Bayes and Support Vector Machines respectively. For a given name, the methods learn a certain model from the train data and use the model to predict whether a new citation is authored by the author. However, the method is user-dependent. It is impractical to train thousands of models for all individuals in a large digital library. In contrast to supervised methods, our method is more scalability.

The other type of related work is semi-supervised clustering, e.g. [3] [7]. [3] proposes a probabilistic model for semi-supervised clustering based on Hidden Markov Random Fields. Their model combines the constraint-based and distance-based approaches. Compared with [3], we define six kinds of constraints and our method generates the constraints automatically.

6. Conclusion

In this paper, we have investigated the problem of researcher social network extraction, an important issue for mining social networks. We have formalized the extraction problem. We have then proposed a unified approach to perform the profile extraction task and a constraint-based probabilistic model to perform name disambiguation in integration. Experimental results show that our approaches outperform the baseline methods on both of the two issues. When applying it to expert finding, we obtain a significant improvement on performances.

References

[1] H. Alani, S. Kim, D. Millard, M. Weal, W. Hall, P. Lewis, and N. Shadbolt, "Automatic Ontology-Based Knowledge Extraction from Web Documents", *IEEE Intelligent Systems*, 2003, 18(1):14-21.

[2] K. Balog and M. Rijke, "Finding Experts and their Details in E-mail Corpora", In *Proc. of WWW'2006.*

[3] S. Basu, M. Bilenko, and R.J. Mooney, "A Probabilistic Framework for Semi-Supervised Clustering", In *Proc. of KDD'2004*, Seattle, USA, August 2004, pp. 59-68.

[4] R. Bekkerman and A. McCallum, "Disambiguating Web Appearances of People in a Social Network", In *Proc. of WWW'2005*, ACM Press, 2005, pp. 463-470.

[5] D. Brickley and L. Miller, "FOAF Vocabulary Specification, Namespace Document", September 2, 2004. http://xmlns.com/foaf/0.1/.

[6] F. Ciravegna. $(LP)^2$, "An Adaptive Algorithm for Information Extraction from Web-related Texts", In *Proc. of the IJCAI-2001 Workshop on Adaptive Text Extraction and Mining*, Seattle, USA, August 2001.

[7] D. Cohn, R. Caruana, and A. McCallum, "Semi-supervised Clustering with User Feedback", *Technical Report* TR2003-1892, Cornell University, 2003

[8] M. Collins, "Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms", In *Proc. of EMNLP'2002.*

[9] C. Cortes and V. Vapnik, "Support-Vector Networks", *Machine Learning*, 20, 1995:273-297.

[10] N. Craswell, A. de Vries, and I. Soboroff, "Overview of the Trec-2005 Enterprise Track", *TREC 2005 Conference Notebook*, pp.199-205

[11] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan, "GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications", In *Proc. ACL'2002.*

[12] Z. Ghahramani and M. Jordan, "Factorial Hidden Markov Models", *Machine Learning*, 1997, 29:245-273.

[13] L. Gillick and S. Cox, "Some Statistical Issues in the Comparison of Speech Recognition Algorithms", In *Proc. of ICASSP'1989*, Vol. 1: 532-535

[14] J. Hammersley and P. Clifford, "Markov Fields on Finite Graphs and Lattices", *Unpublished manuscript*, 1971.

[15] H. Han, L. Giles, H. Zha, C. Li, and K. Tsioutsiouliklis, "Two Supervised Learning Approaches for Name Disambiguation in Author Citations", In *Proc. of JCDL'2004,* Tucson, Arizona, USA, June 2004, pp. 296–305.

[16] H. Han, H. Zha, and C.L. Giles, "Name Disambiguation in Author Citations using a K-way Spectral Clustering Method", In *Proc. of JCDL'2005,* pp. 334–343.

[17] T. Kristjansson, A. Culotta, P. Viola, and A. McCallum, "Interactive Information Extraction with Constrained Conditional Random Fields", In *Proc. of AAAI'2004.*

[18] J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data", In *Proc. of ICML'2001.*

[19] McCallum, D. Freitag, and F. Pereira, "Maximum Entropy Markov Models for Information Extraction and Segmentation", In *Proc. of the ICML'2000.*

[20] E. Minkov, W.W. Cohen, and A.Y. Ng, "Contextual Search and Name Disambiguation in Email Using Graphs", In *Proc. of SIGIR'06*, Washington, USA, 2006, pp. 27-34.

[21] R. Sproat, A. Black, S. Chen, S. Kumar, M. Ostendorf, and C. Richards, "Normalization of non-standard words", *WS'99 Final Report*, 1999

[22] Y.F. Tan, M. Kan, and D. Lee, "Search Engine Driven Author Disambiguation", In *Proc. of JCDL'2006*, Chapel Hill, NC, USA, June 2006, pp. 314-315.

[23] J. Tang, H. Li, Y. Cao, and Z. Tang, "Email Data Cleaning", In *Proc. of SIGKDD'2005*, pp. 489-499.

[24] J. Tang, M. Hong, D. Zhang, B. Liang, and J. Li, "Information Extraction: Methodologies and Applications", *Emerging Technologies of Text Mining: Techniques and Applications*, Hercules A. Prado and Edilson Ferneda (Ed.), Idea Group Inc., Hershey, USA, 2007.

[25] C.J. van Rijsbergen, "Information Retrieval", *Butterworths*, London, 1979.

[26] E. Xun, C. Huang, and M. Zhou, "A Unified Statistical Model for the Identification of English baseNP", In *Proc. of ACL'2000*, Hong Kong, October 2000.

[27] K. Yu, G. Guan, and M. Zhou, "Resume Information Extraction with Cascaded Hybrid Model", In *Proc. of ACL'2005.*