A Combination Approach to Web User Profiling

Jie Tang
Tsinghua University
Limin Yao*
University of Massachusetts Amherst
Duo Zhang*
University of Illinois at Urbana-Champaign
and
Jing Zhang
Tsinghua University

In this paper, we study the problem of Web user profiling, which is aimed at finding, extracting, and fusing the 'semantic'-based user profile from the Web. Previously, Web user profiling was often undertaken by creating a list of keywords for the user, which is (sometimes even highly) insufficient for main applications. This paper formalizes the profiling problem as several subtasks: profile extraction, profile integration, and user interest discovery. We propose a combination approach to deal with the profiling tasks. Specifically, we employ a classification model to identify relevant documents for a user from the Web and propose a Tree-structured Conditional Random Fields (TCRF) to extract the profile information from the identified documents; we propose a unified probabilistic model to deal with the name ambiguity problem (several users with the same name) when integrating the profile information extracted from different sources; finally, we use a probabilistic topic model to model the extracted user profiles, and construct the user interest model. Experimental results on an online system show that the combination approach to different profiling tasks clearly outperforms several baseline methods. The extracted profiles have been applied to expert finding, an important application on the Web. Experiments show that the accuracy of expert finding can be improved (ranging from +6% to +26% in terms of MAP) by taking advantage of the profiles.

Categories and Subject Descriptors: H.3.3 [**Information Search and Retrieval**]: Text Mining; H.4.0 [**Information Systems Applications**]: General; H.2.8 [**Database Management**]: Database Applications

General Terms: Algorithms, Experimentation

Additional Key Words and Phrases: User Profiling, Information Extraction, Name Disambiguation, Topic Modeling, Social Network

* The work was done when the second and third authors were studying in Tsinghua University.
Corresponding author's address: Jie Tang, Rm 1-308, FIT Building, Tsinghua University, Beijing, 100084. China.

1. INTRODUCTION

Profiling of a Web user is the process of obtaining values of different properties that constitute the user model. Considerable efforts have been made to mine the user's interests from his/her historical data. A typical way for representing the user's interests is to create a list of relevant keywords. However, such a profile is *insufficient* for modeling and understanding users' behaviors. A complete user profile (including one's education, experience, and contact information) is very important for providing high quality Web services. For example, with a well-organized user profile base, online advertising can be more targeted based on not only user's interests but also his/her current position.

Traditionally, user profiling was viewed as an engineering issue and was conducted manually or undertaken separately in a more or less ad-hoc manner. For instance, in web-based social networks such as MySpace and YouTube, the user has to enter the profile by her/him-self. Unfortunately, the information obtained solely from the user entering profile is sometimes incomplete or inconsistent. Users do not fill some information merely because they are not willing to fill the information.

Some other work builds the user profile with a list of keywords generated using statistical methods, for example using high frequent words discovered from the user-entered information or user-browsed Web pages. However, such a method ignores some important semantic information such as location and affiliation.

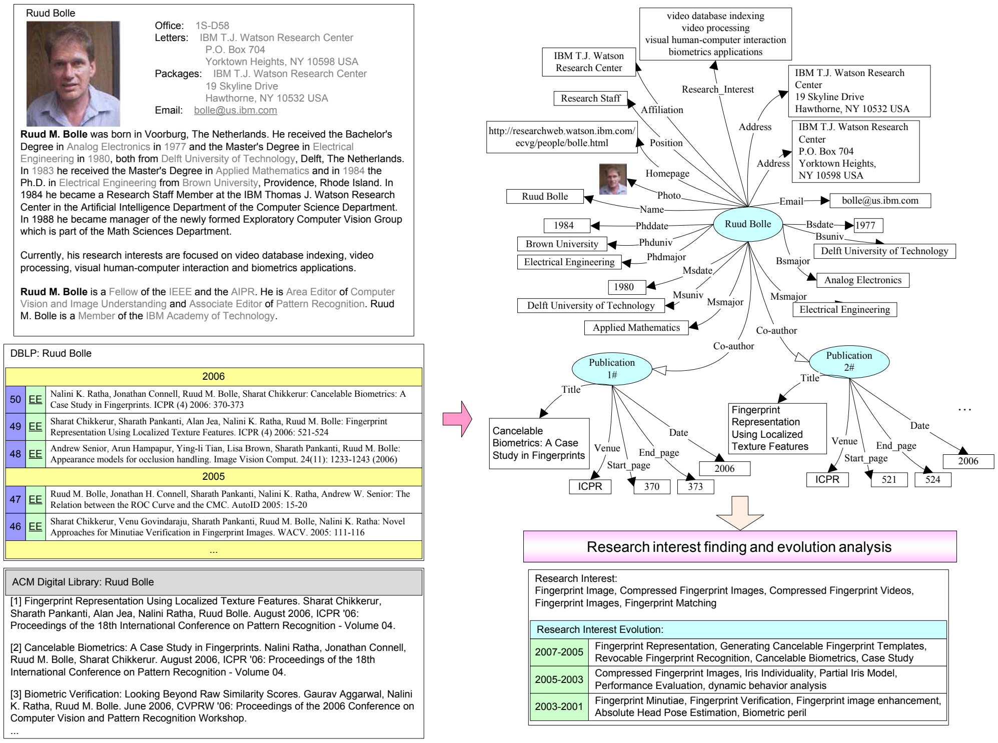
Recently, a few works have been conducted to automatically build the semantic-based user profile using information extraction technologies [Alani et al. 2003] [Pazzani and Billsus 1997] [Yu et al. 2005]. Most of the existing methods use predefined rules or specific machine learning models to extract the different types of profile information in a separated fashion. However, some profile information (e.g., user interests) is implied in the user related documents (e.g., blogs) and cannot be explicitly extracted from the Web page.

1.1 Motivating Example

To clearly motivate this work, we demonstrate with an example drawn from a real-world system, ArnetMiner (`http://www.arnetminer.org/`). In this system, one basic goal is to create a profile for each researcher, which contains basic information (e.g. photo, affiliation, and position), contact information (e.g. address, email, and telephone), educational history (e.g. graduated university and major), research interests, and publications. For each researcher, some of the profile information can be extracted from his/her homepage or Web pages introducing him/her; some other profile information (e.g., publications) should be integrated from online digital libraries (e.g., DBLP or ACM); and the other information (e.g., research interests) should be mined from the collected information.

Figure 1 shows an example of researcher profile. The left part shows the researcher's homepage and his DBLP/ACM page which contains his publication papers. The ideal profiling results are shown in the right part of Figure 1. The right-bottom part shows the researcher's interests mined from the publication papers.

Such a profiling result can benefit many data mining and social network applications. For example, if all researchers' profiles are correctly created, we will have



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
...		

ACM Digital Library: Ruud Bolle

[1] Fingerprint Representation Using Localized Texture Features. Sharat Chikkerur, Sharath Pankanti, Alan Jea, Nalini Ratha, Ruud Bolle. August 2006, ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition - Volume 04.

[2] Cancelable Biometrics: A Case Study in Fingerprints. Nalini Ratha, Jonathan Connell, Ruud M. Bolle, Sharat Chikkerur. August 2006, ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition - Volume 04.

[3] Biometric Verification: Looking Beyond Raw Similarity Scores. Gaurav Aggarwal, Nalini K. Ratha, Ruud M. Bolle. June 2006, CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop.
...

Research interest finding and evolution analysis

Research Interest:
Fingerprint Image, Compressed Fingerprint Images, Compressed Fingerprint Videos, Fingerprint Images, Fingerprint Matching

Research Interest Evolution:	
2007-2005	Fingerprint Representation, Generating Cancelable Fingerprint Templates, Revocable Fingerprint Recognition, Cancelable Biometrics, Case Study
2005-2003	Compressed Fingerprint Images, Iris Individuality, Partial Iris Model, Performance Evaluation, dynamic behavior analysis
2003-2001	Fingerprint Minutiae, Fingerprint Verification, Fingerprint image enhancement, Absolute Head Pose Estimation, Biometric peril

Fig. 1. An example of researcher profiling.

a large collection of well-structured database about researchers in the world. We can use the profiles to help with mining applications such as expert finding, which aims to find experts on a given topic.

The challenges of user profiling are as follows: (1) How to identify relevant pages for a given user and how to extract the profile information from the identified pages? (2) How to integrate the profiles extracted from different sources/pages, as the profile information of a user might be distributed on multiple pages? (3) How to discover user interests implied in the user associated documents?

For extraction of the profile, the manual entering mean for each user 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 user. However, most of existing methods employed a predefined rule or a specific machine learning model to separately identify each property of the profile. However, it is *highly ineffective* to use the separated methods to do profile extraction due to the natural disadvantages of the methods: (1) For each property in the profile, one has to define a specific rule or a specific 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 extracted from different sources, we need to deal with the name ambiguity problem (several users with the same name). Existing methods include heuristic rules, classification-based supervised method, and clustering-based unsupervised method. However, it is ineffective to directly employ the existing methods in user 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 other persons; and (3) The clustering-based unsupervised method cannot use the dependencies between papers and also cannot use the supervised information.

For discovery of user interests, it is also insufficient to use the existing keyword-based methods. There are two main reasons: (1) These methods do not consider the semantic relationship between words; and (2) The methods ignore the dependencies between users, for example users who co-author many papers may have the same interests.

1.2 Our Solution

In this paper, we aim to conduct a systematic investigation of the problem of Web user profiling. First, we decompose Web user profiling as three subtasks: profile extraction, name disambiguation, and user interest discovery. All of the three subtasks can be formalized using graphical models. Specifically, for profile extraction, as the information on the Web is naturally laid-out in a hierarchical structure, we propose formalizing the problem in a tree-structured conditional random fields. For name disambiguation, the problem is to assign papers to different persons with a same name. We formalize the problem in a Markov random graph, where each node denotes a paper and edge denotes relationship (e.g., coauthor) between papers. For user interest discovery, we propose a generative graphical model, where the paper writing procedure is formalized in a series of probabilistic steps. To the best of our knowledge, our work is the first to formalize all the subtasks of user profiling in a combination approach and tackle all the problems at once.

We have implemented the proposed approaches in the system ArnetMiner.org. The system has been in operation on the internet for more than three years and has attracted user accesses from 190 countries. In total, more than half million researchers' profiles have been extracted. We conduct experiments for extracting researchers' profiles. Experimental results indicate that our method clearly outperforms the methods of using separated models for profile extraction. Experimental results also indicate that our disambiguation method can outperform existing methods. We apply the proposed methods to expert finding. Experimental results show that our methods of profile extraction, name disambiguation, and user interest analysis can indeed enhance expert finding (+26% in terms of MAP).

Our contributions in this paper include: (1) a formalization of the problem of user profiling, (2) a proposal of a unified tagging approach to extract user profile, (3) a proposal of a probabilistic method to name disambiguation, (4) a proposal of a topic

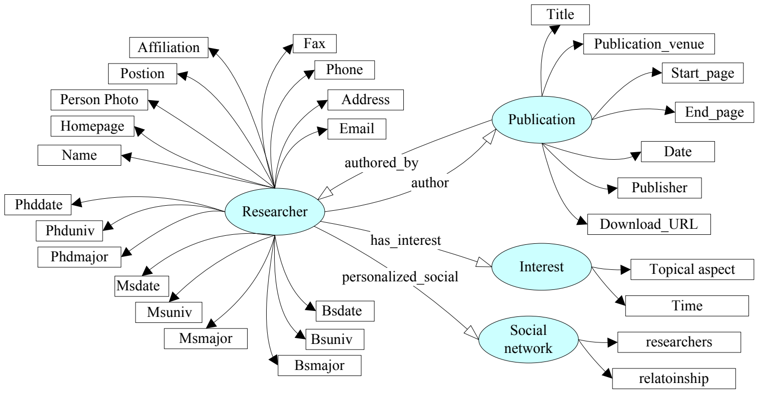


Fig. 2. Schema of researcher profile.

model to perform topical analysis of user interests, and (5) an empirical verification of the effectiveness of the proposed approaches. The approaches proposed in this paper are general and can be applied to many applications, e.g., social network extraction and information integration.

The rest of the paper is organized as follows. In Section 2, we formalize the problem of Web user profiling. In Section 3, we give an overview of our approaches. In Section 4, we explain our approach to profile extraction and in Section 5 we describe how we deal with the name ambiguity problem when integrating the extracted profiles. In Section 6 we present our method for user interests discovery. Section 7 gives the experimental results. Section 8 describes a demonstration system. Finally, before concluding the paper in Section 10, we introduce related work.

2. PROBLEM FORMULATION

In different applications, definitions of profile schemas might be different. In this paper, we use the researcher profile as the example for explanation. The definition of the researcher profile and the proposed approaches for user profiling can be easily extended to other applications.

We define the schema of the researcher profile (as shown in Figure 2), by extending the FOAF ontology [Brickley and Miller 2004]. In the schema, 4 concepts, 29 properties and 4 relations are defined. The social network denotes the subsocial graph related to the current researcher. The interest denotes the semantic topical aspect, which will be detailed later. The publication denotes documents co-authored by the researcher.

We use the data from the ArnetMiner system for study. The system tries to provide a social networking platform for academic researchers. It has gathered 648,289 researcher profiles. Our statistical study 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 name ambiguity problem (different researchers with the same name), we have examined 100 random selected researcher names and found that more than 30% of the names have the ambiguity problem.

We here describe the three key issues we are going to deal with: profile extraction, name disambiguation, and user interests.

(1) Profile extraction. We produced statistics on randomly selected 1,000 researchers. We observed that 85.6% of the researchers are faculties of universities and 14.4% are from company research centers. For researchers from a 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 varies largely depending on the authors. We have also found that 71.9% of the 1,000 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.

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 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 $1,325$ cases (14.5%) in our data that property labels of the tokens need use the extraction results of the other tokens. An ideal method should consider processing all the subtasks together.

Moreover, different from previous data extraction work, information on the Web page is usually organized hierarchically. For example, in the researcher homepage of Figure 1, the top information block contains the basic information (e.g. a photo, two addresses, and an email address), the middle block describes the educational history information (e.g., graduate universities and majors), and the bottom block includes the professional services information (e.g., position and affiliation information). An immediate observation is that identification of the type of the information block would be greatly helpful to identify the information contained in the block.

(2) Name disambiguation. We do not perform extraction of publications directly from one's homepage. 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 $1,200,000$ 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 name ambiguity problem.

We give a formal definition of the name disambiguation task in our context. Given a person name a, we denote all publications having the author name a as $P = \{p_1, p_2, \cdots, p_n\}$. For u authors of a paper $\{a_i^{(0)}, a_i^{(1)}, \cdots, a_i^{(u)}\}$, we call the author name we are going to disambiguate as the principal author (denoted as $a_i^{(0)}$) and the others secondary authors. Suppose there are k actual researchers having the name a, our task is then to assign papers with the name a to their actual researcher y_h, $h \in [1, k]$.

(3) User interests. We do not extract research interests directly from the researchers' homepages, 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 the homepages; secondly, research interest is usually implied by the associated documents, e.g., papers published by the researcher.

Formally, we define user interest on the basis of topics. Each topic is defined as $z = \{(w_1, p(w_1|z)), \cdots, (w_{N1}, p(w_{N1}|z))\}$. The definition means that a topic is represented by a mixture of words and their probabilities belonging to the topic. The topic definition can be also extended to other information sources. For example, in the academic application, we can extend the topic definition by publication venues c, i.e., $z = \{(c_1, p(c_1|z)), \cdots, (c_{N1}, p(c_{N1}|z))\}$. Finally, the interests of researcher a is defined as a set of topic distributions $\{P(z|a)\}_z$.

3. THE OVERVIEW OF OUR APPROACH

We propose a combination approach to solve the user profiling problem. Figure 3 shows the overview of our approach. There are mainly two components: profile extraction and integration, and user interest analysis. The first component targets at extracting and integrating profile information from the Web; while the second component targets at analyzing users' interests.

In the profile extraction and integration component, given a researcher name, we first use the Google API to retrieve a list of documents that contain the researcher name. Then we employ a classification model to identify whether a document in the list is the homepage or an introducing page of the researcher. Next, we use an extraction model to extract the profile information from the identified pages. In particular, we view the problem as that of assigning tags to the input texts, with each tag representing a profile property.

We crawl the publication information from several online digital libraries (e.g., DBLP). We integrate the publication information and extracted profile information. We propose a probabilistic model to deal with the name ambiguity problem for integrating the extracted user profiles. The model can incorporate any types of domain background knowledge or supervised information (e.g., user feedbacks) as features to improve the performance of disambiguation.

In the user interest analysis component, we use a probabilistic topic model to discover the latent topic distribution associated with each researcher. Then we use the discovered topic distributions as the researcher interests.

In this paper, our main technical contributions lie in the approaches we propose to deal with the three subtasks in the two components: profile extraction, integration, and user interest discovery. Theoretically, all the three approaches are based on probabilistic graphical model. More specifically, for profile extraction and integration, our approaches are based on the theory of Markov Random Field [Hammersley and Clifford 1971]. Markov Random Field (MRF) is a probability distribution of labels (hidden variables) that obeys the Markov property. It can be formally defined as follows.

Definition 3.1. 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.

The proposed model for profile extraction is a Tree-structured Conditional Random Fields (TCRFs) and the proposed model for name disambiguation is based

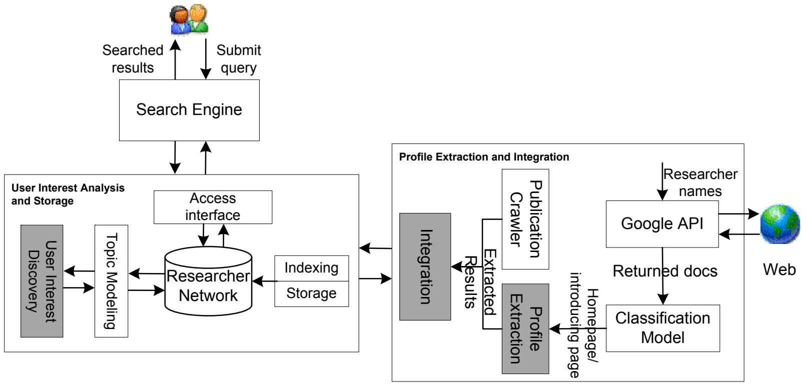


Fig. 3. Approach overview.

on Hidden Markov Random Fields (HMRFs) [Basu et al. 2004]. The reasons we use the two models are: (1) Such a model can describe the dependencies between information, thus can improve the accuracy of profile extraction and name disambiguation. (2) For profile extraction, we can label some training data for supervised learning; while for name disambiguation it is difficult to provide sufficient training data. Therefore, we propose using a discriminative model (TCRF) for profile extraction and a generative model (HMRF) for the name disambiguation task. (3) Both models can be easily extended, thus for different applications we can extend the model based on application-specific features.

As for user interest analysis, the proposed model is a multi-level Bayesian network, which models each paper by following a stochastic process: first one of the paper's authors would decide what topic z to write according to his/her research interest (i.e. topic distribution) $\{P(z|a)\}_z$. Then a word w_{di} is sampled from the topic z according to the word distribution of the topic $\{P(w|z)\}_w$. This series of probabilistic steps can capture well the process of authors writing a paper. In addition, parameters (topic distribution and word distribution) can be estimated in a unsupervised way. Another reason of using the Bayesian network for user interest analysis is that we can easily incorporate different types of objects (e.g., researchers, publication venues, and papers) into one model, thus we can uncover the latent dependencies between the heterogeneous objects.

In the follow sections, we will describe the proposed approaches in more detail.

4. PROFILE EXTRACTION

4.1 Process

There are three steps: relevant page finding, preprocessing, and tagging. In relevant page finding, given a researcher name, we first get a list of web pages by a search engine (i.e. Google) and then identify the homepage or introducing page using a binary classifier. We use support Vector Machines (SVM) [Cortes and Vapnik 1995] as the classification model and define features such as whether the title of the page contains the person name and whether the URL address (partly) contains the person name. The performance of the classifier is 92.39% by F1-measure.

In preprocessing, (a) we segment 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 or a tree-structure of units in the tagging problem. In tagging, given a sequence of units or a tree-structure of units, we determine the most likely corresponding tags using a trained tagging model. Each tag corresponds to a property defined in Figure 2. In this paper, we present a Tree-structure Conditional Random Fields (TCRFs) [Tang et al. 2006] as the tagging model. Next we describe the steps (a) and (b) in detail.

(a) We identify tokens in the Web page heuristically. 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 [Sproat et al. 2001] include email address, IP address, URL, date, number, percentage, words containing special symbols (e.g. 'Ph.D.', 'Prof.'), unnecessary tokens (e.g. '===' and '###'), etc. We identify special words using regular expressions. '< image >' tokens are '< image >' tags in the HTML file. We identify them by parsing the HTML file. Terms are base noun phrases extracted from the Web pages. We employed the methods proposed in [Xun et al. 2000]. Punctuation marks include period, question, and exclamation mark.

(b) We assign tags to each token based on their corresponding type. For standard word, we assign all possible tags. 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* (it is likely that an email address is shown as an image). For term token, we assign *Position*, *Affiliation*, *Address*, *Bsmajor*, *Msmajor*, *Phdmajor*, *Bsuniv*, *Msuniv*, and *Phduniv*. In this way, each token can be assigned several possible tags. Using the tags, we can perform extraction of 16 profile properties, which cover 95.71% of the property values on the Web pages).

4.2 Extraction Model using Conditional Random Fields

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 [Lafferty et al. 2001]. However, the previous linear-chain CRFs only model the linear-dependencies as a sequence, but is not able to model hierarchical dependencies [Lafferty et al. 2001] [Zhu et al. 2006].

In this section, we first introduce the basic concepts of Conditional Random Fields (CRFs) and the linear-chain CRFs, and then we explain a Tree-structured CRF (TCRF) model to model the hierarchically laid-out information. Finally we discuss how to perform parameter estimation and extraction in TCRFs.

4.2.1 *Linear-chain CRFs.*

Conditional Random Fields are undirected graphical models [Lafferty et al. 2001]. As defined before, X is a random variable over data sequences to be labeled, and Y is a random variable over corresponding label sequences. All components Y_i of Y are assumed to range over a finite label alphabet Y. CRFs construct a conditional model $P(Y|X)$ with a given set of features from paired observation and label sequences.

A CRF is a random field globally conditioned on the observation X. Linear-chain

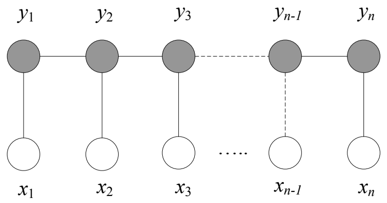


Fig. 4. Graphical representation of Linear-chain CRFs.

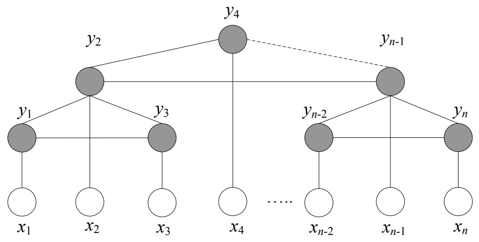


Fig. 5. Graphical representation of Tree-structured CRFs.

CRFs were first introduced by Lafferty et al [Lafferty et al. 2001]. An example graphical structure of linear-chain CRFs is shown in Figure 4.

By the fundamental theorem of random fields [Hammersley and Clifford 1971], the conditional distribution of the labels y given the observations data x has the form

$$P(y|x) = \frac{1}{Z(x)} exp(\sum_{e \in E,j} \lambda_j t_j(e, y|_e, x) + \sum_{v \in V,k} \mu_k s_k(v, y|_v, x)) \tag{1}$$

where x is a data sequence, y is a label sequence, and $y|_e$ and $y|_v$ are the set of components of y associated with edge e and vertex v in the linear chain respectively; t_j and s_k are feature functions; parameters λ_j and μ_k correspond to the feature functions t_j and s_k respectively, and are to be estimated from the training data; $Z(x)$ is the normalization factor, also known as partition function.

4.2.2 *Tree-structured Conditional Random Fields (TCRFs).* Linear-chain CRFs cannot model dependencies across hierarchically laid-out information. We propose a Tree-structured Conditional Random Field (TCRF) model [Tang et al. 2006]. The graphical structure of TCRFs is shown in Figure 5.

From Figure 5, we see that y_4 is the parent vertex of y_2 and y_{n-1} (for simplifying description, hereafter we use parent-vertex to represent the upper-level vertex and use child-vertex to represent the lower-level vertex). TCRFs can model the parent-child dependencies, e.g. $y_4 - y_2$ and $y_4 - y_{n-1}$. Furthermore, y_2 and y_{n-1} are in the same level, which are represented as a sibling dependency in TCRFs.

Here we also use X to denote the random variable over observations, and Y to denote the corresponding labels. Y_i is a component of Y at the vertex i. Same

Table I. Definition of information block and profile properties.

Block Type	**Profile Property**
Photo	Person photo
Basic information	Position, Affiliation
Contact information	Fax, Phone, Address, Email
Educational history	Phddate, Phduniv, Phdmajor, Msdate, Msuniv, Msmajor, Bsdate, Bsuniv, Bsmajor

as the linear-chain CRFs, we consider one vertex or two vertices as a clique in TCRFs. TCRFs can be also viewed as a finite-state model. Each variable Y_i has a finite set of state values and we assume the one-to-one mapping between states and labels. Thus dependencies across components Y_i can be viewed as transitions between states.

$$P(y|x) = \frac{1}{Z(x)} exp(\sum_{c \in C,j} \lambda_j t_j(c, y|_c, x) + \sum_{v \in V,k} \mu_k s_k(v, y|_v, x)) \tag{2}$$

where c is a clique defined on edge (e.g., parent-child (y_p, y_c), child-parent (y_c, y_p), and sibling edge (y_s, y_s)) or triangle (e.g., (y_p, y_s, y_s)). t_j and s_k are feature functions.

TCRFs have the same form as that of linear-chain CRFs except that in TCRFs the edges include parent-child edges, child-parent edges, and sibling-vertices edges, while in linear-chain CRFs the edges mean the transitions from the previous-state to the current-state.

In researcher profile extraction, the observation x in TCRFs corresponds to the identified homepage/introducing page. The tree is obtained by converting the Web page into a DOM tree. The root node denotes the Web page, each leaf node in the tree denotes the word token, and the inner node denotes the coarse information block (e.g., a block containing contact information). The label y of the inner node thus corresponds one type of the coarse information block; while the label y of the leaf node corresponds to one of the profile properties. Definitions of the researcher profile properties and the coarse information block, as well their relationships are summarized in Table I.

4.2.3 *Parameter Estimation.* The parameter estimation problem is to determine the parameters $\Theta = \{\lambda_1, \lambda_2, \cdots; \mu_k, \mu_{k+1}, \cdots\}$ from training data $D = \{(x^{(i)}, y^{(i)})\}$ with empirical distribution. More specifically, we optimize the log-likelihood objective function with respect to a conditional model $P(y|x, \Theta)$:

$$L_\Theta = \sum_i \tilde{P}(x^{(i)}, y^{(i)}) \text{log} P_\Theta(x^{(i)}, y^{(i)}) \tag{3}$$

In the following, to facilitate the description, we use f to denote both the edge feature function t and the vertex feature function s; use c to denote both edge e and vertex v; and use λ to denote the two kinds of parameters λ and μ. Thus, the derivative of the objective function with respect to a parameter λ_j associated with

(1) **Initialization**: for every node u and every pair of nodes (u, v), initialize T_u^0 by $T^0 = \kappa\psi_u$ and $T_{uv}^0 = \kappa\psi_{uv}$, with κ being a normalization factor.

(2) **TRP Updates**: for $i = 1, 2, \cdots$, do:
- —Select some spanning tree Γ^i with edge set E^i, where $R = \{\Gamma^i\}$ is a set of spanning trees;
- —Use any exact algorithm, such as belief propagation, to compute exact marginals $P^i(x)$ on Γ^i. For all $(u, v) \in E^i$, set

$$T_u^{i+1}(x_u) = P^i(x_u),\ T_{uv}^{i+1}(x_u, x_v) = \frac{P^i(x_u, x_v)}{P^i(x_u)P^i(x_v)};$$

- —Set $T_{uv}^{i+1} = T_{uv}^i$ for all $(u, v) \in E \setminus E^i$ (i.e. all the edges not included in the spanning tree Γ^i);
- —Stop if termination conditions are met.

Fig. 6. The TRP algorithm.

clique index c is:

$$\frac{\partial L_\Theta}{\partial \lambda_j} = \sum_i \left[\sum_c f_j(c, y_{(c)}^{(i)}, x^{(i)}) - \sum_y \sum_c P(y_{(c)}|x^{(i)}) f_j(c, y_{(c)}^{(i)}, x^{(i)}) \right] \tag{4}$$

where $y_{(c)}^i$ is the label assignment to clique c in $x^{(i)}$, and $y_{(c)}$ ranges over label assignments to the clique c. We see that for each clique, we need to compute the marginal probability $P(y_{(c)}|x^{(i)})$. The marginal probability $P(y_{(c)}|x^{(i)})$ can be again decomposed into: $P(y_p, y_c|x^{(i)})$, $P(y_c, y_p|x^{(i)})$, $P(y_s, y_s|x^{(i)})$, and $P(y_i|x^{(i)})$, as we have three types of dependencies and one type of vertex. Moreover, we need to compute the global conditional probability $p(y^{(i)}|x^{(i)})$.

The marginal probabilities can be done using many inference algorithms for undirected model (for example, Belief Propagation [Yedidia et al. 2001]). However, as the graphical structure in TCRFs can be a tree with cycles, exact inference is infeasible. We propose using the Tree-based Reparameterization (TRP) algorithm [Wainwright et al. 2001] to compute the approximate probabilities of the factors. TRP is based on the fact that any exact algorithm for optimal inference on trees actually computes marginal distributions for pairs of neighboring vertices. For an undirected graphical model over variables x, this results in an alternative parameterization of the distribution as:

$$P(x) = \frac{1}{Z} \prod_{s \in V} \psi_s(x_s) \prod_{(s,t) \in E} \psi_{st}(x_s, x_t) \Rightarrow P(x) = \prod_{s \in V} P_s(x_s) \prod_{(s,t) \in E} \frac{P_{st}(x_s, x_t)}{P_s(x_s)P_t(x_t)} \tag{5}$$

where $\psi_s(x_s)$ is the potential function on single-vertex x_s and $\psi_{st}(x_s, x_t)$ is the potential function on edge (x_s, x_t); and Z is the normalization factor.

TRP consists of two main steps: Initialization and Updates. The updates are a sequence of $T_n \rightarrow T_{n+1}$ on the undirected graph with edge set E, where T represents the set of marginal probabilities maintained by TRP including single-vertex marginals $T_u^{n+1}(x_u)$ and pairwise joint distribution $T_{uv}^{n+1}(x_u, x_v)$; and n denotes the iteration number. The TRP algorithm is summarized in Figure 6.

So far, the termination conditions in the TRP algorithm are defined as: if the maximal change of the marginals is below a predefined threshold or the update times exceed a predefined number (defined as $1,000$ in our experiments), then stop the updates. When selecting spanning trees $R = \{\Gamma^i\}$, the only constraint is that the trees in R cover the edge set of the original undirected graph U. In practice, we select trees randomly, but we always first select edges that have never been used in any previous iteration.

Finally, to reduce overfitting, we define a spherical Gaussian weight prior $P(\Theta)$ over parameters, and penalize the log-likelihood object function as:

$$L_\Theta = \sum_i P(x^{(i)}, y^{(i)}) \log P_\Theta(x^{(i)}, y^{(i)}) - \frac{\|\lambda\|^2}{2\sigma^2} + const \tag{6}$$

with gradient

$$\frac{\partial L_\Theta}{\partial \lambda_j} = \sum_i \left[\sum_c f_j(c, y^{(i)}_{(c)}, x^{(i)}) - logz(x^{(i)}) \right] - \frac{\lambda_j}{\sigma^2} \tag{7}$$

where *const* is a constant.

The function L_Θ is convex, and can be optimized by any number of techniques, as in other maximum-entropy models [Lafferty et al. 2001]. In the result below, we used gradient-based L-BFGS [Liu et al. 1989], which has previously outperformed other optimization algorithms for linear-chain CRFs [Sha and Pereira 2003].

4.2.4 *Extraction.* Extraction (also called as ‘labeling’) is the task to find labels y^* that best describe the observations x, that is, $y^* = \max_y P(y|x)$. Dynamic programming algorithms, the most popular methods for this problem, can be used for extraction in TCRFs. We use the DOM tree presented in the Web page to infer the hierarchical structure. Then we use the TRP algorithm to compute the maximal value of $p(y|x)$.

4.2.5 *Features.* For each token unit, three types of features are defined: content features, pattern features, and term features.

1. Content Features

For a standard word, the content features include:

—Word features. Whether the current token is a standard word.

—Morphological features. The morphology of the current token, e.g. whether the token is capitalized.

For a ‘< image >’ token, the content features include:

—Image size. The size of the current image.

—Image height/width ratio. The ratio of the height to the width of the current image. The ratio of a person photo is likely to be greater than 1.0.

—Image format. The format of the image (e.g. “JPG”, “BMP”).

—Image color. 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).

—Face recognition. Whether the current image contains a person face. We used a face recognition tool (`http://opencvlibrary.sf.net`) to detect the person face.

—Image filename. Whether the image filename (partially) contains the researcher name.

—Image “ALT”. Whether the “alt” attribute of the “< image >” tag (partially) contains the researcher name.

—Image positive keywords. Whether the image filename contains positive keywords like “myself”.

—Image negative keywords. Whether the image filename contains negative keywords like “logo”.

2. Pattern Features Pattern features are defined for each token.

—Positive words. Whether the current token contains positive *Fax/Phone* keywords like “Fax:”, “Phone:”, positive *Position* keywords like “Manager”.

—Special tokens. Whether the current token is a special word.

3. Term Features Term features are defined only for term token.

—Term features. Whether the token unit is a term.

—Dictionary features. Whether the term is included in a dictionary.

We can easily incorporate these features into our model by defining Boolean-valued feature functions. Finally, two sets of features are defined in the CRF model: transition features and state features. For example, a transition feature $y_{i-1} = y'$, $y_i = y$ implies that if the current tag is y and the previous tag is y', then the value is true; otherwise false. The state feature $w_i = w$, $y_i = y$ implies that if the token is w and the current tag is y, then the feature value is true; otherwise false. In total, $308,409$ features were used in our experiments.

5. NAME DISAMBIGUATION

We crawled the publication data from existing online data sources. 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 name ambigity problem.

The goal of name disambiguation is to disambiguate n papers $P = \{p_1, p_2, \cdots, p_n\}$ that contain the author name a to k actual researchers $\{y_1, y_2, \cdots, y_k\}$ with respect to name a, i.e., assigning an author label to each paper.

We propose a probabilistic model to deal with the problem. Our intuition in this method is based on two observations: (1) papers with similar content tend to have the same label (belonging to the same author); and (2) papers that have strong relationship tend to have the same labels, for example, two papers are written by the same coauthors.

Our method is based on Hidden Markov Random Field (HMRF) model, a special case of MRF. The reason we chose HMRF is due to its natural advantages. First, like all MRF family members, HMRF can be used to model dependencies (or relationships, e.g., CoAuthor) between observations (each paper is viewed as an observation). Second, HMRF supports unsupervised learning, supervised learning, and also semi-supervised learning. In this paper, we will focus on unsupervised learning for name disambiguation using HMRF, but it is easy to incorporate some

Table II. Relationships between papers.

R	W	Relation Name	Description
r_1	w_1	Co-Pubvenue	$p_i.pubvenue = p_j.pubvenue$
r_2	w_2	Co-Author	$\exists r, s > 0, a_i^{(r)} = a_j^{(s)}$
r_3	w_3	Citation	p_i cites p_j or p_j cites p_i
r_4	w_4	Constraints	Feedbacks supplied by users
r_5	w_5	$\tau-$CoAuthor	$\tau-$extension co-authorship ($\tau > 1$)

prior/supervised information into the model, thus extend the proposed approach to semi-supervised learning. Third, it is natural to do model selection in the HMRF model. The objective function in the HMRF model is a posterior probability of hidden variables given observations, which can be used as a criterion for model selection.

In the rest of this section, we will introduce the hidden Markov Random Field model and then define the objective function for the name disambiguation problem.

5.1 Data Preparation

Each publication p_i has six attributes: paper title ($p_i.title$), publication venue ($p_i.pubvenue$), publication year ($p_i.year$), abstract ($p_i.abstract$), authors ($\{a_i^{(0)}, a_i^{(1)}, \cdots, a_i^{(u)}\}$), and references ($p_i.references$). 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.

We define five types of relationships between papers (Table II). Relationship r_1 represents two papers are published at the same venue. Relationship r_2 means two papers have a same secondary author, and relationship r_3 means one paper cites the other paper. Relationship r_4 indicates a constraint-based relationship supplied via user feedbacks. For instance, the user may specify that two papers should be disambiguated to the same author. We use an example to explain relationship r_5. Suppose p_i has authors "David Mitchell" and "Andrew Mark", and p_j has authors "David Mitchell" and "Fernando Mulford". We are going to disambiguate "David Mitchell". If "Andrew Mark" and "Fernando Mulford" also coauthor another paper, then we say p_i and p_j have a 2-CoAuthor relationship.

Specifically, to test whether two papers have a $\tau-$CoAuthor relationship, we construct a Boolean-valued matrix M, in which an element is 1 if its value is greater than 0; otherwise 0 (cf. Figure 7). In matrix M, $\{p_1, p_2, \cdots, p_n\}$ are publications with the principle author name a. $\{a_1, a_2, \cdots, a_p\}$ is the union set of all $p_i.authors \backslash a_i^{(0)}$, $i \in [1, n]$. Note that $\{a_1, a_2, \cdots, a_p\}$ does not include the principle author name $a_i^{(0)}$. Sub matrix M_p indicates the relationship between $\{p_1, p_2, \cdots, p_n\}$ and initially it is an identity matrix. In sub matrix M_{pa}, an element on row i and column j is equal to 1 if and only if $a_j \in p_i.authors$, otherwise 0. The matrix M_{ap} is symmetric to M_{pa}. Sub matrix M_a indicates the co-authorship among $\{a_1, a_2, \cdots, a_p\}$. The value on row i and column j in M_a is equal to 1 if and only if a_i and a_j coauthor one paper in our database (not limited in $\{p_1, p_2, \cdots, p_n\}$), otherwise 0. Then $\tau-$CoAuthor can be defined based on $M^{(\tau+1)}$, where $M^{(\tau+1)} = M^{(\tau)}M$ with $\tau > 0$.

$$
\begin{array}{c|cccc|ccc|}
 & p_1 & p_2 & \dots & p_n & a_1 & \dots & a_p \\
p_1 & 1 & 0 & \dots & 0 & 1 & \dots & 0 \\
p_2 & 0 & 1 & & & 0 & & 1 \\
\dots & \dots & & & & & & \\
p_n & 0 & & & 1 & 1 & & 0 \\
\hline
a_1 & 1 & 0 & & 1 & 1\,0 & & 1 \\
\dots & \dots & & & & & & \\
a_p & 0 & 1 & & 0 & 1\,0 & & 1
\end{array}
\qquad
\begin{array}{c|c|c|}
 & p_1\ p_2\ \dots\ p_n & a_1\ \dots\ a_p \\
\begin{array}{c} p_1 \\ p_2 \\ \dots \\ p_n \end{array} & M_p & M_{pa} \\
\hline
\begin{array}{c} a_1 \\ \dots \\ a_p \end{array} & M_{ap} & M_a
\end{array}
$$

Fig. 7. Matrix M for r_5 relationship.

The publication data with relationships can be modeled as a graph comprising of nodes and edges. Attributes of a paper are represented as a feature vector. In the vector, we use words (after stop words filtering and stemming) in the attributes of a paper as features and use occurring times as the values.

5.2 Formulation using Hidden Markov Random Fields

Hidden Markov Random Fields (HMRF) is a member of the family of MRF and its concept is derived from Hidden Markov Models (HMM) [Ghahramani and Jordan 1997]. A HMRF is mainly composed of three components: an observable set of random variables $X = \{x_i\}_{i=1}^n$, a hidden field of random variables $Y = \{y_i\}_{i=1}^n$, and neighborhoods between each pair of variables in the hidden field.

We formalize the disambiguation problem as that of grouping relational papers into different clusters. Let the hidden variables Y be the cluster labels on the papers. Every hidden variable y_i takes a value from the set $\{1, \cdots, k\}$, which are the indexes of the clusters. The observation variables X correspond to papers, where every random variable x_i is generated from a conditional probability distribution $P(x_i|y_i)$ determined by the corresponding hidden variable y_i.

Figure 8 shows an example graphical representation of HMRF. The observation variable x_i corresponds to a paper and the hidden variable y_i corresponds to the assignment result. The dependent edge between the hidden variables corresponds to the relationship between papers (cf. Table II for the definition of the relationship).

By the fundamental theorem of random fields [Hammersley and Clifford 1971], the probability distribution of the label configuration Y has the form:

$$P(Y) = \frac{1}{Z_1}\exp(\sum_{(y_i,y_j)\in E,k} \lambda_k f_k(y_i, y_j)) \tag{8}$$

and we assume the publication data is generated under the spherical Gaussian distribution, thus we have:

$$P(X|Y) = \frac{1}{Z_2}\exp(\sum_{x_i \in X,l} \alpha_l f_l(y_i, x_i)) \tag{9}$$

where $f_k(y_i, y_j)$ is a non-negative potential function (also called feature function) defined on edge (y_i, y_j) and E represents all edges in the graph; $f_l(y_i, x_i)$ is a potential function defined on node x_i; λ_k and α_l are weights of the edge feature function and the node potential (feature) function respectively; Z_1 and Z_2 are normalization factors (also called partition functions).

Fig. 14. An example researcher profile.

9. RELATED WORK

9.1 User Profiling

There are two types of research work on user profiling: profile extraction and profile learning.

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 [Yu et al. 2005]. In the first pass, a resume is segmented into consecutive blocks attached with labels indicating the information type. And in the second pass, the detailed information such as Address and Email are identified in certain blocks. The Artequakt system [Alani et al. 2003] uses a rule based extraction system called GATE [Cunningham et al. 2002] to extract entity and relation information from the Web. Michelson and Knoblock propose a unsupervised method to extract 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.

A few efforts also 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 [Kristjansson et al. 2004]. See also [Balog et al. 2006]. Contact information extraction is a subtask of profile extraction, thus it significantly differs from the profile extraction.

Many information extraction models have been proposed. Hidden Markov Model (HMM) [Ghahramani and Jordan 1997], Maximum Entropy Markov Model (MEMM) [McCallum et al. 2000], Conditional Random Field (CRF) [Lafferty et al. 2001], Support Vector Machines (SVM) [Cortes and Vapnik 1995], and Voted Perceptron [Collins 2002] are widely used models. Sarawagi and Cohen [Sarawagi and Cohen 2004] also propose a semi-Markov Conditional Random Fields for information extraction. However, most of the existing models do not consider the hierarchically laid-out structure on the Web. [Tang et al. 2007] gives an overview of the existing literatures on information extraction.

The other type of research is to learn the user profile from user associated documents or user visiting logs. For example [Pazzani and Billsus 1997] discusses algorithms for learning and revising user profiles that can determine which World Wide Web sites on a given topic would be interesting to a user. It uses a Naive Bayes classifier to incrementally learn profiles from user feedback on the Web sites. [Chan 1999] has developed a personalized web browser. It learns a user profile, and aims at helping user navigating the Web by searching for potentially interesting pages for recommendations. [Soltysiak and Crabtree 1998] describes an experimental work to study whether user interests can be automatically classified through heuristics. The results highlighted the need for user feedbacks and machine learning methods.

9.2 Name Disambiguation

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

For example, [Bekkerman and McCallum 2005] tries to distinguish Web pages to different individuals 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 clustering method. The methods are based on unsupervised clustering and cannot describe the relationships between data points.

There are also many works focusing on name disambiguation on publication data. For example, Han et al. propose an unsupervised learning approach using K-way spectral clustering method [Han et al. 2005]. They calculate a Gram matrix for each name data set and apply K way spectral clustering algorithm to the Gram matrix to get the result. On and Lee [On and Lee 2007] propose a scalable algorithm for the name disambiguation problem. They adapt the multi-level graph partition technique to solve the large-scale name disambiguation problem. Their algorithm can have a magnitude improvement in terms of efficiency. Bhattacharya and Getoor [Bhattacharya and Getoor 2007] propose a relational clustering algorithm that uses both attribute and relational information for disambiguation. See also [Tan et al. 2006]. This type of method usually uses a parameter-fixed distance metric in their clustering algorithm, while parameters of our distance metric can be learned during the disambiguation.

Two supervised methods are proposed in [Han et al. 2004] based on Naive Bayes and Support Vector Machines. For a given author name, the methods learn a specific model from the train data and use the model to predict whether a new paper is authored by an author with the name. 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. [Basu et al. 2004] [Cohn et al. 2003] [Zhang et al. 2007a]. [Basu et al. 2004] proposes a probabilistic model for semi-supervised clustering based on Hidden Markov Random Fields. Their model combines the constraint-based and distance-based approaches.

9.3 Topic Modeling

Much effort has been made for investigating topic model or latent semantic structure discovery.

Probabilistic latent semantic indexing (pLSI) is proposed by Thomas Hofmann [Hofmann 1999]. The difference between LSA and pLSI is that the latter is based on the likelihood principle and defines a proper generative model of the data; hence it results in a more solid statistical foundation. However, the pLSI has the problem of overfitting and not being able to estimate documents outside of the training set.

Blei et al. introduce a new semantically consistent topic model, Latent Dirichlet Allocation (LDA) [Blei et al. 2003]. The basic generative process of LDA closely resembles pLSI except that in pLSI, the topic mixture is conditioned on each document and in LDA, the topic mixture is drawn from a conjugate Dirichlet prior that remains the same for all documents.

Some other works have also been made for modeling author interests and document content simultaneously. For example, the Author model (also termed as Multi-label Mixture Model) [McCallum 1999] is aimed at modeling the author interests with a one-to-one correspondence between topics and authors. [Rosen-Zvi et al. 2004] presents an Author-Topic model, which integrates the authorship into the topic model and thus can be used to find a topic distribution over document and a mixture of the distributions associated with authors.

McCallum et al. have studied several other topic models in social network analysis [McCallum et al. 2007]. They propose the Author-Recipient-Topic (ART) model, which learns topic distributions based on emails sent between people.

Compared with above topic modeling work, in this paper, we aim at using a unified model (author-conference-topic model) to characterize the topic distributions of multiple inter-dependent objects in the academic social network.

10. CONCLUSION AND FUTURE WORK

In this paper, we have investigated the problem of Web user profiling. We have formalized the profiling problem as several sub tasks. We have proposed a combination approach to deal with the problems. Specifically, we have proposed a Tree-structured Conditional Random Field (TCRF) to extract the profile information from the Web page and proposed a probabilistic model to solve the name ambiguity problem for integrating the profile information from different sources. Further, we have proposed a topic model to discover user interests. Experimental results indicate that our proposed methods outperform the baseline methods. Experiments of expert finding also show that the extracted user profiles can be used to improve the accuracy of expert finding. We have developed a demonstration system based on the proposed approaches. User feedbacks and system logs show that users of the system consider the system is useful.

There are several potential enhancements of this work. First, a general Web page may contain a lot of noise, how to extract accurate profile information from the noisy data is a challenging issue. Second, the performance of name disambiguation can be further improved by incorporating other relationships or human background knowledge. Third, the proposed method for user interest discovery is a unsupervised method and does not consider any domain knowledge. In practice, for a specific domain (e.g., computer science), people may already build some taxonomy (e.g., the ACM categories) to describe the subfields in the domain, which can be used to guide the discovery of user interests.

There are also many other future directions of this work. It would be interesting to investigate how to extract the profile based on partially labeled data. Data labeling for machine learning is usually tedious and time-consuming. How to reduce the labeling work is a challenging problem. It would also be interesting to investigate the dynamic problem. The profile of a researcher might change after years, for example, moved to a new company. Furthermore, in-depth analysis of the user profiles is also important.

11. ACKNOWLEDGMENTS

The work is supported by the Natural Science Foundation of China (No. 60703059, No. 60973102), Chinese National Key Foundation Research (No. 60933013), National High-tech R&D Program (No. 2009AA01Z138), and Chinese Young Faculty Research Fund (No. 20070003093).

REFERENCES

Alani, H., Kim, S., Millard, D. E., Weal, M. J., Hall, W., Lewis, P. H., and Shadbolt, N. R. 2003. Automatic ontology-based knowledge extraction from web documents. *IEEE Intelligent Systems 18,* 1, 14–21.

Andrieu, C., de Freitas, N., Doucet, A., and Jordan, M. I. 2003. An introduction to mcmc for machine learning. *Machine Learning 50*, 5–43.

Baeza-Yates, R. and Ribeiro-Neto, B. 1999. *Modern Information Retrieval.* ACM Press.

Balog, K., Azzopardi, L., and de Rijke, M. 2006. Formal models for expert finding in enterprise corpora. In *Proceedings of the 29th ACM SIGIR International Conference on Information Retrieval (SIGIR'2006)*. 43–55.

Basu, S., Bilenko, M., and Mooney, R. J. 2004. A probabilistic framework for semi-supervised clustering. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'04)*. 59–68.

Bekkerman, R. and McCallum, A. 2005. Disambiguating web appearances of people in a social network. In *Proceedings of the 14th International Conference on World Wide Web (WWW'05)*. 463–470.

Bhattacharya, I. and Getoor, L. 2007. Collective entity resolution in relational data. *ACM Transactions on Knowledge Discovery from Data 1,* 1 (March), 1–36.

Blei, D. M., Ng, A. Y., and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research 3*, 993–1022.

Brickley, D. and Miller, L. 2004. Foaf vocabulary specification. In *Namespace Document, http://xmlns.com/foaf/0.1/*.

Buckley, C. and Voorhees, E. M. 2004. Retrieval evaluation with incomplete information. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'04)*. 25–32.

Cai, D., He, X., and Han, J. 2007. Spectral regression for dimensionality reduction. In *Technical Report. UIUCDCS-R-2007-2856, UIUC.*

Cao, Y., Li, H., and Li, S. 2003. Learning and exploiting non-consecutive string patterns for information extraction. Tech. Rep. MSR-TR-2003-33, Microsoft Research.

Chan, P. K. 1999. A non-invasive learning approach to building web user profiles. In *KDD-99 Workshop on Web Usage Analysis and User Profiling.*

Ciravegna, F. 2001. An adaptive algorithm for information extraction from web-related texts. In *Proceedings of the IJCAI-2001 Workshop on Adaptive Text Extraction and Mining.*

Cohn, D., Caruana, R., and McCallum, A. 2003. Semi-supervised clustering with user feedback. In *Technical Report TR2003-1892, Cornell University.*

Collins, M. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing (EMNLP'02)*. 1–8.

Cortes, C. and Vapnik, V. 1995. Support-vector networks. *Machine Learning 20*, 273–297.

Craswell, N., de Vries, A. P., and Soboroff, I. 2005. Overview of the trec-2005 enterprise track. In *TREC 2005 Conference Notebook*. 199–205.

Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. 2002. Gate: A framework and graphical development environment for robust nlp tools and applications. In *Proceedings of the 40nd Annual Meeting of the Association for Computational Linguistics (ACL'02)*.

Ghahramani, Z. and Jordan, M. I. 1997. Factorial hidden markov models. *Machine Learning 29,* 2-3, 245–273.

Griffiths, T. L. and Steyvers, M. 2004. Finding scientific topics. In *Proceedings of the National Academy of Sciences (PNAS'04)*. 5228–5235.

Hammersley, J. M. and Clifford, P. 1971. Markov field on finite graphs and lattices. *Unpublished manuscript*.

Han, H., Giles, L., Zha, H., Li, C., and Tsioutsiouliklis, K. 2004. Two supervised learning approaches for name disambiguation in author citations. In *Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries (JCDL'04)*. 296–305.

Han, H., Zha, H., and Giles, C. L. 2005. Name disambiguation in author citations using a k-way spectral clustering method. In *Proceedings of the 5th ACM/IEEE-CS joint conference on Digital libraries (JCDL'05)*. 334–343.

Hofmann, T. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd International Conference on Research and Development in Information Retrieval (SIGIR'99)*. 50–57.

Kristjansson, T., Culotta, A., Viola, P., and McCallum, A. 2004. Interactive information extraction with constrained conditional random fields. In *AAAI'04*.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning (ICML'01)*. 282–289.

Liu, D. C., Nocedal, J., and C, D. 1989. On the limited memory bfgs method for large scale optimization. *Mathematical Programming 45*, 503–528.

McCallum, A. 1999. Multi-label text classification with a mixture model trained by em. In *Proceedings of AAAI'99 Workshop on Text Learning*.

McCallum, A., Freitag, D., and Pereira, F. C. N. 2000. Maximum entropy markov models for information extraction and segmentation. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML'00)*. 591–598.

McCallum, A., Wang, X., and Corrada-Emmanuel, A. 2007. Topic and role discovery in social networks with experiments on enron and academic email. *Journal of Artificial Intelligence Research (JAIR) 30*, 249–272.

Mimno, D. and McCallum, A. 2007. Expertise modeling for matching papers with reviewers. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD'07)*. 500–509.

Minka, T. 2003. Estimating a dirichlet distribution. In *Technique Report, http://research.microsoft.com/ minka/papers/dirichlet/*.

Newman, D., Asuncion, A., Smyth, P., and Welling, M. 2007. Distributed inference for latent dirichlet allocation. In *Proceedings of the 19th Neural Information Processing Systems (NIPS'07)*.

On, B.-W. and Lee, D. 2007. Scalable name disambiguation using multi-level graph partition. In *Proceedings of the SIAM Int'l Conf. on Data Mining (SDM'07)*.

Pazzani, M. J. and Billsus, D. 1997. Learning and revising user profiles: The identification of interesting web sites. *Machine Learning 27,* 3, 313–331.

Rosen-Zvi, M., Griffiths, T., Steyvers, M., and Smyth, P. 2004. The author-topic model for authors and documents. In *Proceedings of the 20th International Conference on Uncertainty in Artificial Intelligence (UAI'04)*. 487–494.

Sarawagi, S. and Cohen, W. W. 2004. Semi-markov conditional random fields for information extraction. In *Proceedings of the 17th Neural Information Processing Systems (NIPS'04)*. 1185–1192.

Sha, F. and Pereira, F. 2003. Shallow parsing with conditional random fields. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL'03)*. 134–141.

Soltysiak, S. J. and Crabtree, I. B. 1998. Automatic learning of user profiles — towards the personalisation of agent services. *BT Technology Journal 16,* 3, 110–117.

Sproat, R., Black, A. W., Chen, S., Kumar, S., Ostendorf, M., and Richards, C. 2001. Normalization of non-standard words. *Computer Speech Language*, 287–333.

Steyvers, M., Smyth, P., and Griffiths, T. 2004. Probabilistic author-topic models for information discovery. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD'04)*. 306–315.

Tan, Y. F., Kan, M.-Y., and Lee, D. 2006. Search engine driven author disambiguation. In *Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries (JCDL'06)*. 314–315.

Tang, J., Hong, M., Li, J., and Liang, B. 2006. Tree-structured conditional random fields for semantic annotation. In *Proceedings of the 5th International Semantic Web Conference (ISWC'06)*. 640–653.

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

Tang, J., Jin, R., and Zhang, J. 2008. A topic modeling approach and its integration into the random walk framework for academic search. In *Proceedings of 2008 IEEE International Conference on Data Mining (ICDM'08)*. 1055–1060.

Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., and Su, Z. 2008. Arnetminer: Extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD'08)*. 990–998.

Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. 2004. Hierarhical dirichlet processes. In *Technical Report 653, Department of Statistics, UC Berkeley.*

van Rijsbergen, C. 1979. *Information Retrieval.* But-terworths, London.

Wainwright, M. J., Jaakkola, T., and Willsky, A. S. 2001. Tree-based reparameterization for approximate estimation on loopy graphs. In *Proceedings of the 13th Neural Information Processing Systems (NIPS'01)*. 1001–1008.

Xun, E., Huang, C., and Zhou, M. 2000. A unified statistical model for the identification of english basenp. In *Proceedings of The 38th Annual Meeting of the Association for Computational Linguistics (ACL'00)*. 3–6.

Yedidia, J. S., Freeman, W. T., and Weiss, Y. 2001. Generalized belief propagation. In *Proceedings of the 13th Neural Information Processing Systems (NIPS'01)*. 689–695.

Yin, X., Han, J., and Yu, P. 2007. Object distinction: Distinguishing objects with identical names. In *Proceedings of IEEE 23rd International Conference on Data Engineering (ICDE'2007)*. 1242–1246.

Yu, K., Guan, G., and Zhou, M. 2005. Resume information extraction with cascaded hybrid model. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL'05)*. 499–506.

Zhai, C. and Lafferty, J. 2001. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th ACM SIGIR International Conference on Information Retrieval (SIGIR'01)*. 334–342.

Zhang, D., Tang, J., and Li, J. 2007a. A constraint-based probabilistic framework for name disambiguation. In *Proceedings of the 16th Conference on Information and Knowledge Management (CIKM'07)*. 1019–1022.

Zhang, J., Tang, J., and Li, J. 2007b. Expert finding in a social network. Proceedings of the 12th Database Systems for Advanced Applications (DASFAA'07), 1066–1069.

Zhu, C., Tang, J., Li, H., Ng, H. T., and Zhao, T. 2007. A unified tagging approach to text normalization. In *Proceedings of the 45th annual meeting of the Association for Computational Linguistics (ACL'07)*. 689–695.

Zhu, J., Nie, Z., Wen, J.-R., Zhang, B., and Ma, W.-Y. 2006. Simultaneous record detection and attribute labeling in web data extraction. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'06)*. 494–503.

Received September 2008; revised February 2009; accepted December 2009