Relational User Attribute Inference in Social Media

Quan Fang, Jitao Sang, Changsheng Xu, *Fellow, IEEE*, and M. Shamim Hossain, *Senior Member, IEEE*

***Abstract*—Nowadays, more and more people are engaged in social media to generate multimedia information, i.e., creating text and photo profiles and posting multimedia messages. Such multimodal social networking activities reveal multiple user attributes such as age, gender, and personal interest. Inferring user attributes is important for user profiling, retrieval, and personalization. Existing work is devoted to inferring user attributes independently and ignores the dependency relations between attributes. In this work, we investigate the problem of relational user attribute inference by exploring the relations between user attributes and extracting both lexical and visual features from online user-generated content. We systematically study six types of user attributes: *gender, age, relationship, occupation, interest*, and *emotional orientation*. In view of methodology, we propose a relational latent SVM (LSVM) model to combine a rich set of user features, attribute inference, and attribute relations in a unified framework. In the model, one attribute is selected as the target attribute and others are selected as the auxiliary attributes to assist the target attribute inference. The model infers user attributes and attribute relations simultaneously. Extensive experiments conducted on a collected dataset from Google+ with full attribute annotations demonstrate the effectiveness of the proposed approach in user attribute inference and attribute-based user retrieval.**

***Index Terms*—Attribute relation, latent SVM (LSVM), user attribute inference.**

I. Introduction

ALONG with the explosive prevalence of social media networking, more and more people are engaged in online social media networks. Facebook now reaches 85% of the world's Internet-using population, i.e., 1.26 billion people in total. Social media users are creating and sharing large-scale multimedia information, such as textual posts, photos, videos, etc. For example, YouTube has reported that in every minute there are 100 hours of videos uploaded, with more than 6 billion hours watch every month.[1] Facebook users have contributed to 250 billion photos totally.[2] Such rich multimedia information reveals important clues of user information including biographic facts (e.g., age, gender, relationship), personal interest (e.g., politics, technology, entertainment, sports), occupation information (e.g., researcher, student, software engineer, musicians), emotional orientation (e.g., optimistic, negative), etc. We call these personal information *user attributes*. Inferring such user attributes can benefit many applications in user profiling, information retrieval, personalization and recommendation.

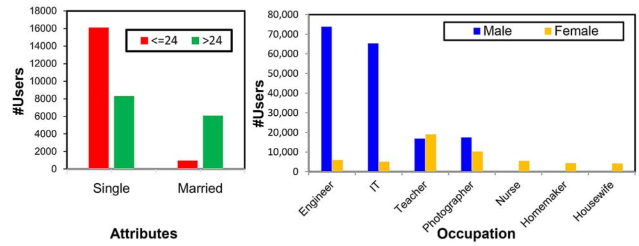

Fig. 1. User attribute co-occurrence statistics on Google+ data.

In most online social networks (e.g., Facebook, Twitter, LinkedIn, and SinaWeibo), such user attributes are not always available. First, users are likely to provide the easy-to-fill basic information such as name, gender, but seldom introduce their interests and other detailed information. Second, due to the privacy issues, most social network sites limit the access to some personal information. Based on our preliminary statistics on the collected Google+ dataset consisting of 19,624 popular users,[3] nearly 90% of the user gender information is provided, while only 12.36% of user birthday and 22.48% of user relationship are obtainable. To address the sparseness of user attributes, several work has been devoted to inferring the latent user attributes such as age, gender, regional origin, location, political orientation, and user-level sentiment by exploiting the online user-generated data [8], [10], [23], [27], [28], [30], [40], [41], showing the potential of user attribute prediction from user social networking activities.

Most existing work has treated user attributes separately and studied user attribute prediction independently. Actually, different types of user attributes have significant dependency relations, e.g., it is not very likely that a 18-year-old man is married and his occupation is most probably student. Fig. 1 shows our dependency analysis results among user attributes of age, gender, relationship and occupation. The statistics are derived based on a total of 105 million Google+ users. In the left sub-

Manuscript received January 26, 2014; revised July 06, 2014; accepted April 16, 2015. Date of publication May 07, 2015; date of current version June 13, 2015. This work was supported in part by the National Basic Research Program of China under Grant 2012CB316304, in part by the National Natural Science Foundation of China under Grant 61225009, Grant 61432019, Grant 61332016, and Grant 61303176, in part by the Beijing Natural Science Foundation under Grant 4131004, and by the Deanship of Scientific Research at King Saud University under the International Research Group Program IRG 14-18. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. K. Selcuk Candan.

Q. Fang, J. Sang, and C. Xu are with the National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: qfang@nlpr.ia.ac.cn; jtsang@nlpr.ia.ac.cn; csxu@nlpr.ia.ac.cn).

M. S. Hossain is with the SWE Department, College of Computer and Information Sciences, King Saud University, Riyadh 12372, Saudi Arabia (e-mail: mshossain@ksu.edu.sa).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2015.2430819

[1][Online]. Available: http://www.youtube.com/yt/press/statistics.html

[2][Online]. Available: http://expandedramblings.com/index.php/by-the-numbers-17-amazing-facebook-stats

[3][Online]. Available: http://socialstatistics.com/top/people

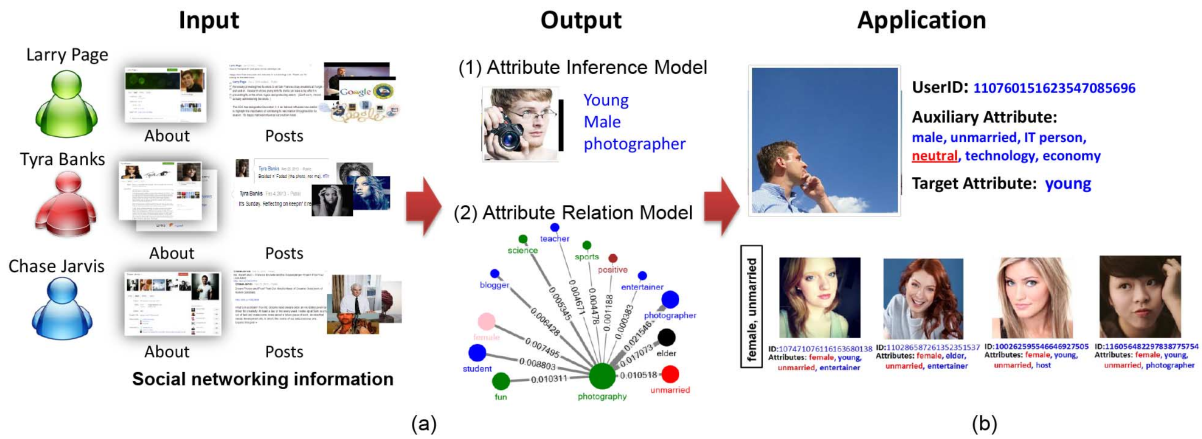

Fig. 2. Problem illustration of relational user attribute inference in social media (best viewed in color). (a) User attribute inference. (b) User profiling and retrieval.

figure, it is conceived that users under 24 years old are more likely to be single. The right sub-figure illustrates that there exist obvious dependencies between the occupation and gender attributes. We argue that the user attribute relation is an important characteristic and will facilitate accurate user attribute inference. Moreover, exploration of attribute relations will enable novel mining and application scenarios. This serves as the basic premise of this paper.[4]

In this paper, we investigate the problem of relational user attribute inference by exploiting the rich user-generated multimedia information and exploring attribute relations in social media network sites. Specially, we study six types of user attributes: *gender, age, relationship, occupation, interest,* and *emotional orientation*. Each type of attribute has multiple values. The two challenges are involved during the attribute inference: First, how to exploit the heterogeneous and multi-modal content for user attribute derivation? Second, how to explore the dependency relations between different types of user attributes for more accurate attribute inference?

We propose a Relational Latent SVM (Relational LSVM) model-based framework to address the two challenges. In particular, we take Google+, the popular social network sites, as the test platform in our study. In Google+, users are allowed to build their profiles on the *About* board and post activities on the *Posts* page. As shown in Fig. 2(a), we formulate the relational user attribute inference problem as follows: The input is user's social networking information including profiles from *About* and posts acclivities from *Posts* in Google+. The Relational LSVM model is developed to learn the output in a supervised discriminative manner, including the predicted user attributes and the inferred attribute relations. In the model, one type of user attribute is selected as *target attribute* and the remained are treated as *auxiliary attributes*. *Target attribute* is the one obtaining direct reinforcement, while the auxiliary attributes are treated as latent variables during the inference. Multiple relations between auxiliary attributes and target attribute are jointly formulated as potential functions into the model. With the derived user attribute and attribute relations, we apply them to applications of user profiling and attribute-based user retrieval as shown in Fig. 2(b). We evaluate the model on a collected real-world dataset with full attribute annotations from Google+. The results demonstrate the effectiveness of our Relational LSVM model for user attribute inference and the potential of attribute relation in user-centric applications. Therefore, the main contributions of this paper can be summarized as follows.

- We address the relational user attribute inference problem by considering user attribute dependency relations in social media. The derived dependency relations contribute to more accurate attribute inference.
- A structured Relational LSVM model is presented to infer user attribute and mine attribute relation compatibility patterns simultaneously. User features, attribute inference, and attribute relations are modeled in a principled way.
- We evaluate the relational attribute inference model with a real-world dataset from Google+. The dataset is completely annotated with various defined attributes. Extensive experiments including user attribute inference and attribute-based user retrieval demonstrate the effectiveness of the proposed approach.

The rest of the paper is structured as follows. Section II reviews the related work. Section III formally define the problem. Section IV gives the data collection and observation. Section V elaborates the proposed model. Experimental results are reported in Section VI. Finally, we conclude the paper in Section VII.

II. Related Work

Recently, user attribute inference has drawn extensive interest from many disciplines. Various user attributes have been investigated by exploiting user contributed data. The first interesting user attribute is *gender*. Researchers have attempted to investigate its identification from conversations [6], [18], blogs [7], [20], [29], user search queries [36], and tweets [28], [40]. Another line is to understand the *political orientation* of users from the congressional transcripts as well as microblogs [9], [10], [27], [28], [31], [40]. Other previously explored user attributes include *geographical location* [8], [14], *regional origin* [28],

[4][Online]. Available: http://www.gplusdata.com/

TABLE I
USER ATTRIBUTE DEFINITION

Attribute Name	Attribute Values
Gender	1-Male; 2-Female
Age	1-Young(≤30); 2-Elder(≥30)
Relationship	1-Unmarried; 2-Married
Occupation	1-Student(St); 2-Information Technology Person (IT), Software Engineer, Geek; 3-Entertainer, Musician, Actor, Comedian, Model, TV show host; 4-Writer, Journalist, Blogger, Editor, TV news host, Critics Lawyer; 5-Politician; 6-Sports star, Athlete; 7-Business man, Economist, Entrepreneur, Market strategist, Financiers; 8-Scientist, Professional, Researcher, Expert; 9-Photographer Traveler; 10-Doctor, Dentist, Pharmacist, Beautician ; 11-Chef, Eater, Cook; 12-Engineer, Specialist, Designer; 13-Teacher; 14-Artist, Religious people, Culture Writer, Designer, Author, Critic; 15-Other
Interest	1-Technology, Information, Internet; 2-News, Politics,military, Society; 3-Economy, Business Manage Strategy; 4-Entertainment, Music, Movie, Fashion; 5-Photography, Travel; 6-Food&Drink; 7-Daily things, Lives life living, Fun interest, Personal Stuff; 8-Sports, Exercise, Body-Building; 9- Thinker, ideas religion culture literature art; 10-Health, Medical care, Treatment, Makeup; 11-Science, Knowledge; 12-Other
Emotional Orientation	1-Positive (fantastic, great, elated, bouncy, jubilant, excited, cheerful, ecstatic); 2-Negative (annoyed, aggravated, bad, pain, embarrassed, bored, anxious, crazy, depressed, scared, sick, angry, sad, score); 3-Neutral (normal, awake, calm, working, blank, report, news, fact)

age [28], [40], *occupation* [17], *ethnicity* [27], *sentiment* [26], [30], [34], etc.

For user attribute inference, there are two types of approaches: rule-based or learning-based.

Several research efforts have been made for directly detecting and extracting user attribute information from text. For example, Garera *et al.* [19] proposed approaches to model structural, transitive and latent properties of biographical free-text data for extracting biographic facts such as *birthdate, occupation, nationality*, etc. Yu *et al.* [39] used a cascaded information extraction framework for identifying personal information from resumes, where resumes are first segmented into certain blocks and profile information such as address are then identified. Zhou *et al.* [42] treated the problem of identifying and extracting biographical sentences as a summarization task. Mann *et al.* [22] aimed to extract specific facts such as *birthplace* using the contextual pattern-learning approach such as "$< \mathrm{NAME} >$ was born in $< \mathrm{Birthplace} >$ or $< \mathrm{NAME} >$ (born) $< \mathrm{Birthplace} >$". Bergsma *et al.* [4] proposed to use conceptual class attributes to predict the hidden demographical properties of social media users. While the above work using the rule-based or pattern learning approaches can effectively extract the user attribute information, the learned patterns or rules are constrained to specific attribute inference and lack of generalization capability.

Most recent work aims to extract user features and learn models for user attribute inference from user communication data and online user-generated multimedia content. User profiles largely impact the characteristics and content of their online communication streams and social networking activities. Zheleva *et al.* [41] and Mislove *et al.* [23] have analyzed the consistency between user profile and social networking activities, and provided statistical proof for inferring user attributes from user online data. Most of the work tackles the user attribute inference problem by designing attribute-specific features and combining with off-the-shelf classifiers. For example, Garera *et al.* [18] extended the N-gram based models proposed in [6] with sociolinguistic features and linear SVM model and showed applicability to a variety spoken conversational transcripts and more formal enron email corpus. Rao *et al.* [28] studied Twitter user attribute detection using a mixture of sociolinguistic features as well as n-gram models. Van Durme *et al.* [13] developed a streaming framework to infer attributes of discourse participants. Pennacchiotti *et al.* [27] attempted to classify users by employing a large set of aggregate features including profile features, tweeting behavior features, linguistic content features, and social network features. Bergsma *et al.* [3] proposed the communication-based name and location clustering method on Twitter to improve user classification. Volkova *et al.* [33] proposed novel approaches for making predictions over dynamically evolving social media streams based on incremental Bayesian online updates. Culotta *et al.* [1] predicted the demographics of Twitter users from website traffic data. The shared limitation of the above work is that, the user attributes are trained and tested independently, where their relations are ignored. As discussed in the introduction, a user exhibits multiple attributes and these attributes correlate significantly with each other. Therefore, in this work, in addition to exploiting a rich set of user features, we aim to provide a novel framework for user attribute inference by taking advantage of the attribute relations.

III. PROBLEM DEFINITION

In this section, we first introduce some necessary definitions and then formulate the problem.

User Attributes: We use user attribute to describe a type of user properties in social media networks. In this work, we consider six types of user attributes including three biographic attributes–*age, gender*, *relationship*, and three personalization attribute–*occupation, interest*, *emotional orientation*. The attribute values are defined manually based on a comprehensive study of Google+ data and a survey of previous work on user attribute inference [17], [28], [40]. Table I presents the meaning of user attribute values.[5] The six types of attributes definition are described as follows.

- *Gender*. Gender is a binary valued attribute. We use gender to describe whether a user is male or female.
- *Age*. Age is a real valued attribute. Given the general lack of ground truth for user age, exact age inference is impossible in social networks. We conduct a detailed

[5]In the following sections, we will mix using "attribute" and "attribute value" when no ambiguity is caused.

investigation and observation on Google+ users. Generally, Google+ users can be categorized into two groups of young and elder. Hence we divide the user into two major demographic pools: users who are below 30 (young) and users who are above 30 (elder). This binary categorization is simple but useful and reasonable for user modeling. Same setting is also used in [28].

- *Relationship.* Relationship in Google+ has multiple categories such as *single, married, in a relationship*, etc. For the sake of clarity, we classify users into two groups of unmarried and married.
- *Occupation.* Based on the study of occupation function on the Google+ users pages and referred to the work in [21], occupation is described with 15 values such as *IT professionals, entertainer, and photographers*, etc.
- *Interest.* Interest refers to the favorite topics based on users' posts. Based on the analysis of our collected Google+ data, we define 12 kinds of topics of interest to cover a large interest category. Since each user may have several interests, the interest value is vector-based and we treat interest inference as a binary classification problem.
- *Emotional Orientation.* Posts of a user can reflect his/her specific emotional status. For example, a user with many interesting and happy posts is very likely to be a positive person, while posts containing negative content indicate the user's negative tendency. Emotional orientation is used to describe the emotional polarity of a user based on his/her posts. We define three emotional orientation values: positive, negative, and neutral.

In our work, user attribute inference is divided into two steps. First, coarse user attributes are derived by training independent classifiers on extracted features from user profiles and posts. Second, we attempt to explore the dependency relations between user attributes to boost the user attribute inference performance. Specifically, we select a type of user attribute (e.g., occupation) as the *target attribute* for which we want to learn a predictive model, and the remained attributes (e.g., age, gender, relationship, etc) called *auxiliary attribute* are used to help learn the model.

Given a collection of Google+ users $\mathcal{U}$, each user $u \in \mathcal{U}$ corresponds to a two dimensional tuple $[\mathcal{X}_u, \mathcal{A}_u]$. $\mathcal{X}_u = [\mathbf{x}_1, \ldots, \mathbf{x}_K]$, where K is the number of attribute types and $\mathbf{x}_k$ is the user feature of the k-th attribute. $\mathcal{A}_u = [a_1, \ldots, a_K]$ denotes the user attribute set. Denote the target attribute as $\mathcal{T}$ and the auxiliary attribute as $\mathcal{S}$. The whole attribute set is denoted as $\mathcal{A} = [\mathcal{S}, \mathcal{T}]$. Thus, the problem is formally defined as:

Problem 1: Relational User Attribute Inference. Given a collection of Google+ users $\mathcal{U}$ and attribute set $\mathcal{A} = [\mathcal{S}, \mathcal{T}]$. Each user $u \in \mathcal{U}$ is defined as a two dimensional tuple $[\mathcal{X}_u, \mathcal{A}_u]$, where $\mathcal{X}_u$ and $\mathcal{A}_u$ are user features and attributes respectively. The goal of relational user attribute inference is to learn (1) a predictive function $f(\mathcal{X}_u, \mathcal{S}) \rightarrow \mathcal{T}_u$ to infer the target attribute label of a user; and (2) attribute relation compatibility $\Psi(a_i, a_k) \in \mathbb{R}^{|\mathcal{A}| \times |\mathcal{A}|}$, where Ψ indicates the compatibility strength of attribute relations.

Our formulation of relational user attribute inference is different from existing work on latent user attribute classification. Instead of only utilizing textual content and studying attributes independently, we propose a structured latent discriminative model to combine multimodal user features and attribute relations for accurate attribute inference.

TABLE II
STATISTICS OF OUR COLLECTED GOOGLE+ DATA

#Users	2,548	#Profile Photos	2,548
#Posts	846,339	#Post Photos	88,988
#Attached Objects	333,331		

TABLE III
NUMBER OF LABELS FOR EACH USER ATTRIBUTE

Attribute	Count
Gender	1,808; 740
Age	728; 1,820
Relationship	1,228; 1,321
Occupation	68; 500; 210; 261; 13; 31; 307; 88; 560; 20; 11; 141; 28; 131; 179
Interest	685; 179; 174; 385; 891; 70; 704; 91; 169; 21; 152; 47;
Emotional Orientation	1,371; 62; 1,115

IV. DATA AND OBSERVATIONS

A. Data Collection

In this study, we collect our experimental dataset from Google+ via its public available API. To download the dataset, we first built a top 20,000 user ID list from Google+ Social Statistics.[6] For each user in the list, we issued his/her ID into the Google+ API and crawled the profile information and recent 500 posts (if there have). The profile text metedata and profile photos of users are crawled. For each post, we downloaded the posted text content and attached objects such as articles, photos, video descriptions. The initial dataset contains 19,624 users. The users with less than 20 posts are removed. We preprocess the data to filter out the non-individual or non-english users. This results in 2,548 users and 846,339 posts. Table II lists statistics of the collected dataset.

As aforementioned, we study six types of user attributes including gender, age, relationship, occupation, interest, and emotional orientation. Since we are interested in predicting the user attributes and there is no available groundtruth of user attribute values, we build the evaluation dataset by manually labeling the attributes for each user. To ease the annotation task, some user platforms such as Facebook,[7] Wikipedia[8] are utilized as referenced sources for accurately annotating the user attributes We invited 8 active social network users as annotators. Three annotators are assigned for each user record. The annotators are asked to refer to substantial information from Facebook, Wikipedia, Google Search, to accomplish the attribute annotation of each user. A label is determined as ground-truth if at least two annotators agree on it. Table III shows the distribution of each attribute.

[6][Online]. Available: http://socialstatistics.com/

[7][Online.] Available: http://www.facebook.com/

[8][Online.] Available: http://www.wikipedia.org/

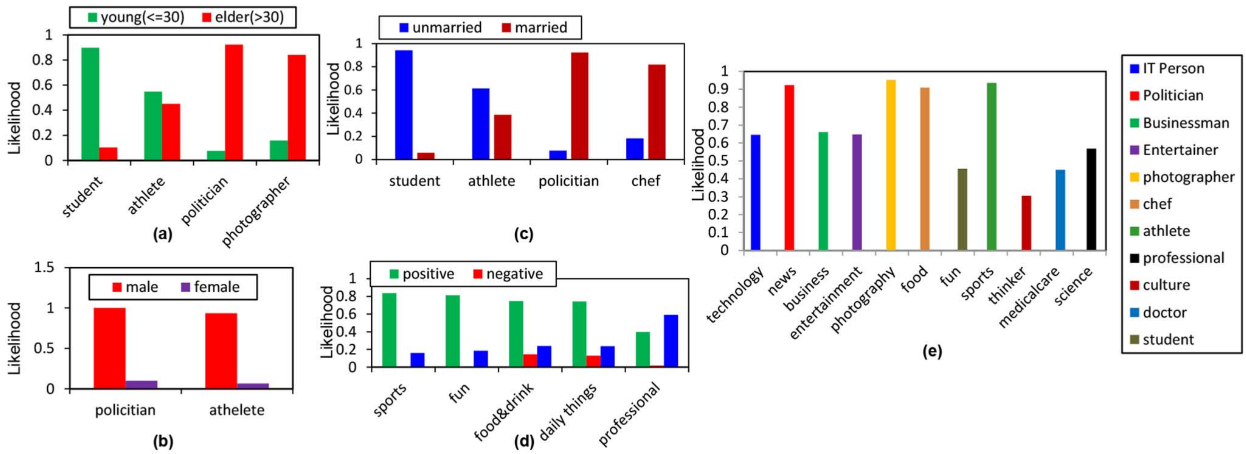

Fig. 3. Attribute correlation in the collected dataset. There exist certain correlations between user attributes. The y-axis represents the likelihood of different user attributes. The subfigures illustrate the correlation likelihood: (a) occupation and age, (b) occupation and gender, (c) occupation and relationship, (d) interest and emotional orientation, and (e) interest and occupation.

B. Observations

Before introducing our solution for the relational user attribute inference problem, we first provide several observations from the annotated dataset. Fig. 3 shows the correlation between user attributes in sub-Fig. 3(a)–(d). For each attribute value a_i shown on the top, we present its most correlated attributes a_k as x-axis labels, according to their co-occurrence likelihood computed as

$$\frac{\#(a_i, a_k)}{min(\#(a_i), \#(a_k))}. \tag{1}$$

We can make some observations. It can be seen that there exist strong correlations between certain attribute value pairs, such as young and student, elder and politician, male and politician, unmarried and student, and positive and fun, etc. Fig. 3(e) illustrates the relation between user occupations and interests (occupation as legend and interest as x-axis label), where strong correlations are proved again. For example, IT users with interest in technology, and politician users with interest in news. These observations verify that there are considerable certain dependency relations between user attributes, which motivates our solution for relational user attribute inference in the following section.

V. Relational User Attribute Inference

In this paper, we first extract a rich set of aggregated features to derive user attributes individually, then we combine all the derived attributes and formalize the relational attribute inference in a structured latent model.

A. User Feature Extraction

We aim to extract a rich set of user features from user-generated multimedia profiles and posts. The user features include six types: sociolinguistic feature, unigram feature, topic-based feature, profile photo feature, profile photo face feature, and post photo feature. Both textual and visual features are considered: The first three are text-based and the latter three are visual content-based.

Textual User Features. For each user, we aggregate his/her profile and all the posts into a big document for textual user feature extraction. Previous work has extensively studied the effectiveness of different textual features on attribute classification [26]–[28]. Generally, unigram and sociolinguistic model with term presence can achieve good results for different attribute inference tasks. These two types of features are both utilized. Sociolinguistic feature is constructed by remaining sociolinguistic words or signs (e.g., umm, uh-huh, $> \_ <, ><$). Unigram model removes these signs to construct the feature. Instead of using all the words in the data collection for feature representation, we use a simple method for discriminative word selection for each attribute category. The basic idea is to measure each word by a score

$$s(t_i) = \left| \frac{\sum_k (ATT_{w_{ki}} - AVE_{w_i})}{N_{w_i}} \right| \cdot \log(N_{w_i}) \tag{2}$$

where $ATT_{w_{ki}}$ is the number of word w_i in the k-th attribute value; AVE_{w_i} and N_{w_i}, respectively, denote the average number and total number of word w_i in the attribute type. We use 10,000 words with the highest scores and the term presence as the feature weight for sociolinguistic feature, and unigram feature construction. Therefore, each user is represented as a 10,000 dimensional binary feature vector. For topic-based feature, LDA [5] is applied to extract latent topics from user profiles and posts. After the topic distillation, each user is represented as the distribution over the derived topic space. In the experiment we use 100 topics and 10,000 words for LDA.

Visual User Features. For visual feature extraction, we consider two types of photos: profile photos and post photos. For profile photo feature, each profile photo is extracted a 809-dimensional feature vector [43], including 81-dimensional color moment, 37-dimensional edge histogram, 120-dimensional wavelet texture feature, 59-dimensional LBP feature [25], and 512-dimensional GIST feature [32]. In addition, most of the profile photos contain faces, which are extremely useful for identifying facial attributes such as age and gender. Therefore, we detect faces from the profile photos and then extract the same 809-dimensional feature vector as the profile photo face

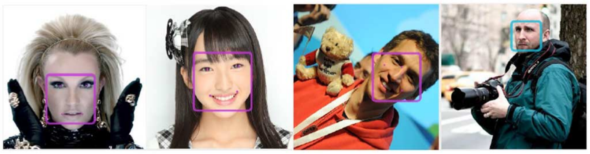

Fig. 4. Some face detection results in the profile photos.

TABLE IV
DETAILED CONCEPT LIST

Concept list
animals beach beauty bird bodypart books building car cartoon cat celebrity child city cloth cloud colors couple crowd dancing dark design dog drink electronic product family flight flower food fruit geek goose grass house icon indoor insect lake landscape leaf man model mountain naturescene office painting palace party people performance photography portrait poster puzzle road room sculpture sea sky snow soldier sport spot squirrel stadium stone store street sunset talk text tiny plant tower toy transport tree universe watch waterfall woman

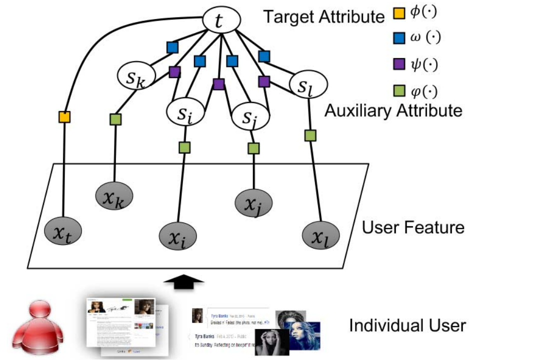

Fig. 5. Illustration of the proposed RLSVM model. Each circle corresponds to a variable, where $\mathbf{x}$, $\mathbf{s}$, and t are user feature vector, auxiliary attributes, and target attribute respectively. The squares represent potential functions including user feature and auxiliary attribute potential $\varphi(\cdot)$, auxiliary attribute and target attribute potential $\omega(\cdot)$, auxiliary attribute and auxiliary attribute potential $\psi(\cdot)$, and user feature and target attribute potential $\phi(\cdot)$.

feature. We use a face detection tool,[9] which can largely handle different face poses and scales in web images. Fig. 4 presents several detection results. The appearances of post photos vary across a wide range of concepts. In order to obtain a compact and semantic representation for each user, we explicitly map each user's post photos onto a pre-defined concept list. The concept list is manually constructed based on the observation of 88,988 downloaded post photos. The reason we construct the concept list by manually defining instead of automatically learning is two-fold. (1) By investigating into post photos, we find that there exist some common concepts, e.g., people, animals, birds, snow, lake, mountain, etc. We can aggregate the photos that describe a common concept for training. (2) The concepts learned automatically capture less semantics and are difficult to be interpreted. We select 81 categories from the collected post photos to construct the concept list, which is shown in Table IV. Each concept category has nearly 100 images for training. We train 81 concept classifiers in a supervised manner. Dense HOG features [11] are extracted for each image and Locality-constrained Linear Coding (LLC) [35] is used to obtain the image representation. For each concept, we train a SVM classifier using LIBLINEAR [15]. The classification confidence is mapped to a probability score by the sigmoid function. Therefore, each photo is finally represented as a 81-dimensional vector corresponding to the concept probability score. Since one user may post several photos, we apply a max-pooling method on the photo representations to obtain an aggregated 81-dimensional feature vector for each user, which is referred as post photo feature.

B. User Attribute Derivation

Based on the extracted user features, we can learn predictive models for user attribute inference. Specifically, for each type of user attributes, we build SVM classifiers with the six types of user features respectively using LIBLINEAR [15]. As a result, there are six SVM classifiers for each specific attribute inference. To combine all the user features for deriving the attribute value, a fusion scheme is needed to combine the confidence score from the six classifiers. We employ a stacked model [37] to perform the fusion. The key idea of stacked model is to learn a meta-level (level-1) classifier based on the outputs of base-level (level-0) classifiers. We utilize another SVM for the task with the concatenated predictions of each classifier from the six types of features. This fusion scheme is simple but well solving our problems of integrating multiple feature-base classifiers.

C. Exploring Attribute Relation for User Attribute Inference

It is well recognized that user attribute correlates with each other when modeled for the same individual. Therefore, we develop a structured discriminative model for incorporating the attribute relations to boost the attribute inference capability. A graphical illustration of the proposed model is shown in Fig. 5. At the lowest level, rich user features are extracted from the multimedia content of an individual user. At the intermediate level, the compatibility between a user's feature vector and attribute label is modeled. At the top level, attribute relations are explored for accurate target attribute inference.

1) Relational Latent SVM Model: We now describe how we model the attributes for an individual user. A user u is denoted as a tuple $(\mathbf{x}, \mathbf{s}, t)$. Here $\mathbf{x}$ is the user feature vector, $\mathbf{s} = (s_1, \ldots, s_L) \in \mathcal{S}$ is the auxiliary attributes, and $t \in \mathcal{T}$ is the target attribute. Our goal is to learn a model that can be used to infer the accurate target attribute label to a test user sample $\mathbf{x}$.

To explore the dependency relations between user attributes, we use an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to model user attributes and capture the attribute relations. A vertex $v_i \in \mathcal{V}$ corresponds to one attribute with certain value, and an edge $(i, k) \in \mathcal{E}$ indicates the relation strength between attribute a_i and attribute a_k. We build the graph $\mathcal{G}$ by examining the co-occurrence statistics of user attribute in the training dataset. The edge weight of attribute a_i and attribute a_k is defined as $\pi(i,k) = \frac{Q(a_i,a_k)}{min(Q(a_i),Q(a_k))}$, where $Q(a_i, a_k)$ denotes the number of co-occurrence between attribute a_i and a_k, and $Q(a_i)$ denotes the number of occurrence for attribute a_i. A large weight $w(i,k)$ means a strong relation between attribute a_i and a_k.

Formally, we are interested in learning a score function $f_{\mathbf{w}} : \mathcal{X} \times \mathcal{L} \rightarrow \mathbb{R}$ over an example user $\mathbf{x}$ and the target attribute t,

[9][Online]. Available: http://www.faceplusplus.com/en/

where $\mathbf{w}$ are the model parameters. Here $\mathcal{X}$ denotes the user feature space. We use $f_w(\mathbf{x}, t)$ to measure the compatibility among the user feature $\mathbf{x}$, the target attribute label t, and the configurations of auxiliary attribute labels h. $f_w(\mathbf{x}, t)$ takes the form of $f_w(\mathbf{x}, t) = \max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}, \mathbf{s}, t)$ to score the confidence of user sample $\mathbf{x}$ labeled as target attribute label t with the auxiliary attribute configuration $\mathbf{s}$, which is defined by combining different potential functions

$$\mathbf{w}^T \mathbf{\Phi}(\mathbf{x}, \mathbf{s}, t) = \alpha^T \phi(\mathbf{x}, t) + \sum_{i \in \mathcal{S}} \beta^T \varphi(\mathbf{x}, s_i) + \sum_{i \in \mathcal{S}} \gamma^T \omega(s_i, t) + \sum_{(i,k) \in \mathcal{E}} \eta^T \psi(s_i, s_k, t). \quad (3)$$

In this model the parameter vector $\mathbf{w}$ is the concatenation of the parameters in all factors. The model simultaneously considers the dependencies among user features and attributes. In particular, the first term predicts the target attribute label from user features; the second term describes the relationship between user features and auxiliary attributes; the third term captures the relationship between auxiliary attributes and target attribute; and the last term represents the dependencies between auxiliary attributes. The details of potential functions in (3) are described in the following.

User Feature vs. Target Attribute Potential $\alpha^T \phi(\mathbf{x}, t)$: This potential is a standard linear model for target attribute prediction. Here $\phi(\mathbf{x}, t)$ represents a certain mapping from user feature $\mathbf{x}$ to attribute label t. In the implementation, we represent $\phi(\mathbf{x}, t)$ as the score of the target attribute SVM classifier learned in Section V-B. Model parameter α re-weights the SVM scores to make the target attribute prediction more accurate. Specifically, the potential is parameterized as

$$\alpha^T \phi(\mathbf{x}, t) = \sum_{b \in \mathcal{T}} \alpha_b \mathbb{1} \cdot (t = b) \cdot g_b \quad (4)$$

where $1(\cdot)$ is the indicator function, g_b is the SVM confidence score corresponding to target attribute b.

User Feature vs. Auxiliary Attribute Potential $\beta^T \varphi(\mathbf{x}, s_i)$: This potential is a standard linear model trained to predict the value of the i-th auxiliary attribute given the user feature $\mathbf{x}$. Similar to the potential in user feature vs. target attribute model, $\varphi(\mathbf{x}, s_i)$ represents the confidence score from the auxiliary attribute SVM classifier learned in Section V-B. The parameter β re-weights the auxiliary attribute SVM scores. This potential function is parameterized as

$$\beta^T \varphi(\mathbf{x}, s_i) = \sum_{c \in \mathcal{S}} \beta_c \cdot \mathbb{1}(s_i = c) \cdot y_c \quad (5)$$

where y_c is the SVM score corresponding to auxiliary attribute c.

Target Attribute vs. Auxiliary Attribute Potential $\gamma^T \omega(s_i, t)$: This potential represents the dependencies between the target attribute t and auxiliary attribute s_i. Model parameter γ encodes the attribute relation compatibility between target attribute and auxiliary attribute. The potential function is defined as

$$\gamma^T \omega(s_i, t) = \sum_{c \in \mathcal{S}} \sum_{b \in \mathcal{T}} \gamma_{c,b} \cdot \mathbb{1}(s_i = c) \cdot \mathbb{1}(t = b) \cdot \pi(c, b) \quad (6)$$

where $\pi(c, b)$ is the edge weight of target attribute b and auxiliary attribute c in the graph $\mathcal{G}$, $\gamma(c, b)$ is the attribute relation compatibility $\Psi(s_i, t)$ between s_i and t.

Auxiliary Attribute vs. Auxiliary Attribute Potential $\eta^T \psi(s_i, s_k, t)$: This potential is used to describe the dependency relations between auxiliary attributes s_i and s_k, which is parameterized as

$$\eta^T \psi(s_i, s_k, t) = \sum_{b \in \mathcal{T}} \sum_{c \in \mathcal{S}} \sum_{d \in \mathcal{S}} \psi_{b,c,d} \cdot \mathbb{1}(t = b) \cdot \mathbb{1}(s_i = c) \cdot \mathbb{1}(s_k = d) \cdot \pi(c, d) \quad (7)$$

where $\pi(c, d)$ is the edge weight of auxiliary attribute c and auxiliary attribute d in the graph $\mathcal{G}$. The parameter $\psi_{b,c,d}$ measures the attribute relation compatibility $\Psi(s_i, s_k, t)$ between auxiliary attribute labels (s_i, s_k) and target attribute label t.

2) Model learning and Inference: We now describe how to learn the model parameters from a set of training data, and how to infer the attribute label given the model parameters.

Given N training examples of users $\mathcal{U} = \{(\mathbf{x}^{(n)}, \mathbf{s}^{(n)}, t^{(n)}\}_{n=1}^{N}$ $(n = 1, 2, \ldots, N)$, we aim to learn the model parameters $\mathbf{w}$ that produce the correct target attribute label t. Note that the auxiliary attributes of the training samples are treated as latent variables and automatically inferred in the model learning process. The information of training samples of auxiliary attributes can be well exploited in the designed potential functions.

We adopt the structured latent SVM formulation [16], [38] to learn the model as follows:

$$\begin{aligned} \min_{w, \xi \geq 0} \quad & \frac{1}{2}\|w\|^2 + C_1 \sum_{n=1}^{N} \xi_n \\ \text{s.t.} \quad & \max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}^{(n)}, \mathbf{s}, t^{(n)}) - \max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}^{(n)}, \mathbf{s}, t) \\ & \geq \Delta(t, t^{(n)}) - \xi_n, \quad \forall n, \forall t \in \mathcal{T} \end{aligned} \quad (8)$$

where C_1 is the trade-off parameter similar to that in SVMs, and ξ_n is the slack variable for the n-th training example to handle soft-margin. Such an objective function requires that the score for ground-truth target attribute label $t^{(n)}$ is much higher than those for other labels. The difference is recorded in a 0-1 loss function $\Delta(t, t^{(n)})$

$$\Delta_{0/1}(t, t^{(n)}) = \begin{cases} 1 & \text{if } t \neq t^{(n)} \\ 0 & \text{otherwise} \end{cases}. \quad (9)$$

The constrained optimization problem in (8) can be equivalently written as an unconstrained problem

$$\begin{aligned} & \min_{\mathbf{w}} L(\mathbf{w}) = \frac{1}{2}\|w\|^2 + C_1 \sum_{i=1}^{N} R_n(\mathbf{w}) \\ & \text{where} R_n(\mathbf{w}) = \max_{t} \left(\Delta_{0/1}(t, t^{(n)}) + \max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}^{(n)}, \mathbf{s}, t) \right) \\ & \qquad - \max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}^{(n)}, \mathbf{s}, t^{(n)}). \end{aligned} \quad (10)$$

We use the non-convex bundle optimization in [12] to solve (10). In a nutshell, the algorithm iteratively builds an increasingly accurate piecewise quadratic approximation of $L(\mathbf{w})$

based on its sub-gradient $\partial_{\mathbf{w}} L(\mathbf{w})$. The key issue is to compute the subgradients $\partial_{\mathbf{w}} L(\mathbf{w})$. We define

$$\begin{aligned} \mathbf{s}^{(n)^*} &= \arg\max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}^{(n)}, \mathbf{s}, t), \forall n, \forall t \in \mathcal{T}, \\ \mathbf{s}^{(n)} &= \arg\max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}^{(n)}, \mathbf{s}, t^{(n)}), \forall n \\ t^{(n)^*} &= \arg\max_{t} (\Delta_{0/1}(t, t^{(n)}) + \max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}^{(n)}, \mathbf{s}, t)) \end{aligned} \tag{11}$$

where $\partial_{\mathbf{w}} L(\mathbf{w})$ can be further computed as

$$\begin{aligned} \partial_{\mathbf{w}} L(\mathbf{w}) = \mathbf{w} &+ C_1 \sum_{i=1}^{N} \mathbf{\Phi}(\mathbf{x}^{(n)}, \mathbf{s}^{(n)^*}, t^{(n)^*}) \\ &- C_1 \sum_{n=1}^{N} \mathbf{\Phi}(\mathbf{x}^{(n)}, \mathbf{s}^{(n)}, t^{(n)}). \end{aligned} \tag{12}$$

Using the subgradients $\partial_{\mathbf{w}} L(\mathbf{w})$, we can optimize (8) using the algorithm in [12] and output the optimal model parameter $\mathbf{w}$.

At each optimization iteration, we also need to infer the latent attribute variables $\mathbf{s}$

$$\mathbf{s}^* = \arg\max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}^{(n)}, \mathbf{s}, t^{(n)}). \tag{13}$$

This is a standard max-inference problem in an undirected graphical model and we use loopy belief propagation [24] to approximately solve it.

D. Applications

After learning the model, we can apply it to a variety of applications. In the following we introduce two potential application scenarios.

1) User Profiling: User profiling is the process of obtaining values of different properties that constitute the user model. Methods for user profiling can typically fall into two categories: profile extraction and profile learning. Our model can be directly applied to user profiling. Given a new user feature $\mathbf{x}$, we can use the model to infer the target attribute and auxiliary attribute as

$$\begin{aligned} t^* &= \arg\max_{t} \left\{ \max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}, \mathbf{s}, t) \right\} \\ \mathbf{s}^* &= \arg\max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}, \mathbf{s}, t^*). \end{aligned} \tag{14}$$

Note that target attribute and auxiliary attribute selection depends on the application scenarios. For example, if we are interested in predicting users's occupation, we select occupation as the target attribute and other attributes as auxiliary ones. The difference between target attribute and auxiliary attribute prediction is that the model can only guarantee more accurate prediction for target attribute while the prediction accuracies for the auxiliary attributes are not guaranteed. The optimization objective function in (8) aims to maximize the margin for target attribute classification. However, we can conduct user profiling by setting each attribute as the target one and run the Relational LSVM model iteratively. After attribute inference, we can construct the user profile and describe the user with multiple attributes.

2) Attribute-Based User Retrieval: Our model can also be used for the retrieval task. Recall that we employ the graph $\mathcal{G}$ to encode the dependency relations for the attribute prediction model. After the model learning, the model parameter $\{\gamma, \eta\} \subset \mathbf{w}$ encodes certain relation compatibility between attributes. The learned user attribute relation compatibility can be utilized to help user retrieval. For example, if we want to search for the users engaged in IT, users who are both male and interested in information technology are more likely to be returned. Based on the model, we extend the attribute-based user retrieval to enable structured queries of one or more attributes and return the most related users corresponding to the specified attributes. Formally, we address this task by solving the following optimization problem:

$$\begin{aligned} \mathbf{x}^* &= \arg\max_{\mathbf{s}} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}, \mathbf{s}, t), \text{given } t. \\ \mathbf{x}^* &= \arg\max_{t} \left\{ \max_{\mathbf{s}'} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}, \mathbf{s}', \hat{\mathbf{s}}, t) \right\}, \text{given } \hat{\mathbf{s}}. \\ \mathbf{x}^* &= \arg\max_{\mathbf{s}'} \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}, \mathbf{s}', \hat{\mathbf{s}}, t), \text{given } t, \hat{\mathbf{s}}. \end{aligned} \tag{15}$$

The first function is to return the users with the highest scores when specifying target attribute t. The second function is to return the users with the highest scores when specifying auxiliary attributes $\hat{\mathbf{s}}$. The third function is to return the users with the highest scores when specifying attributes t and $\hat{\mathbf{s}}$. Note that attribute-based user retrieval, different from user attribute reference, is a structured multi-query retrieval task. Given the query user attributes, the retrieval task is to retrieve the users relevant to the input attributes. The attribute inference task is to predict the most probable attribute labels given the input users. Actually, the structured attribute-based user retrieval is analogous to the Facebook's new darling-Graph Search,[10] which is expected to lead the upcoming trend in social networking discovery.

VI. Experiments

A. Experimental Setting

We qualitatively and quantitatively evaluate the effectiveness of the proposed Relational LSVM in three experiments: attribute relation compatibility analysis, attribute inference, and attribute-based user retrieval. In all experiments, we use 50% of the labeled data for training and the rest for testing.

1) Comparison Methods: We examine the following methods for comparison in user attribute inference and attribute-based user retrieval:

- *SVM-Simple:* the method using six types of user features respectively to train an attribute classifier for one attribute type. The details are described in Section V-B;
- *Stacked SVM:* the method applying stacked SVM to the concatenated confidence scores produced by the attribute classifiers learned with user features. Section V-B presents the detailed description;
- *Relational LSVM:* the proposed method, which incorporates the attribute relation for inference.

2) Evaluation Metrics: We use accuracy to evaluate the attribute inference performance. To evaluate the performance of

[10][Online]. Available: https://www.facebook.com/about/graphsearch

TABLE V
USER FEATURE SPECIFICATIONS AND EXECUTION TIME OF DIFFERENT ATTRIBUTE INFERENCE METHODS

Methods	Feature	Dimensionality	Training Time (seconds)	Test Time (seconds)
SVM-Face	user face	809	$1.056 \sim 5.923$	$0.551 \sim 0.828$
SVM-ProfilePhoto	user profile photo	809	$2.766 \sim 14.19$	$1.162 \sim 1.796$
SVM-PostPhoto	user post photos	81	$0.034 \sim 0.482$	$0.007 \sim 0.065$
SVM-unigram	pure textual words	10,000	$7.887 \sim 58.93$	$3.584 \sim 5.991$
SVM-sociolinguistic	textual and sociolinguistic words	10,000	$6.581 \sim 55.46$	$1.747 \sim 6.313$
SVM-topic-based	textual topics	100	$0.0875 \sim 2.215$	$0.019 \sim 0.264$
Stacked SVM	SVM-simple scores	6	$0.019 \sim 0.347$	$0.006 \sim 0.103$
Relational LSVM	SVM-simple scores and attribute relations	6	$2680 \sim 6620$	$210.5 \sim 394.1$

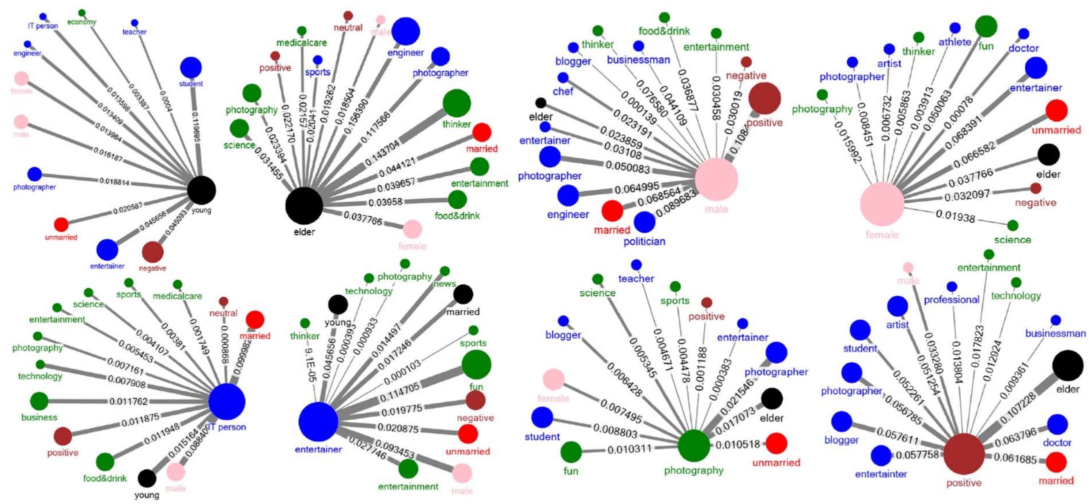

Fig. 6. Visualization of the learned attribute relation compatibility (best viewed in color). The size of balls and links indicates the strength of their correlation.

attribute-based user retrieval, we use Normalized Discounted Cumulative Gain (NDCG), where NDCG at position k is defined as

$$\mathrm{NDCG}@k = \frac{1}{\mathrm{IDCG}} \times \sum_{i=1}^{k} \frac{2^{r_i} - 1}{\log_2(i+1)} \tag{16}$$

where r_i is the relevance rating of item at rank i. For retrieval, r_i is 1 if the user has a link to the returned item and 0 otherwise. IDCG is a normalization constant so that the perfect ranking has a NDCG value of 1.

The learned algorithm for relational SVM model has been implemented in Matlab. All experiments are conducted on a PC running Windows 7 with four Intel Core i5-3470 processors (3.2 GHz) and 4 GB memory. The user feature specifications and execution times of different attribute inference methods on training set and test set are shown in Table V. The execution time of each method for one type of attribute falls into the corresponding time range.

B. Experimental Results and Analysis

1) Attribute Relation Compatibility: Fig. 6 illustrates the visualization of user attribute relation compatibility $\Psi(a_i, a_k) \in \{\gamma, \eta\}$ learned on the training data. The size of the circle and line is proportional to the strength of the relation compatibility. From the figure we can see that the mined attribute relation compatibility results are quite reasonable. We can observe many intuitive co-occurrence patterns, such as young with student, young with unmarried, elder with married, male with politician, female with fun things, positive with married, etc. In particular, personal occupation and interest exhibit strong correlations. For example, IT person has strong links with technology and business; entertainer correlates with entertainment and fun things, photographer strongly links with photography. These meaningful attribute relation patterns derived from the Relational LSVM model suggest that when inferring attribute "photographer" for a user or retrieving users with attribute "photographer", the presence of interest attribute "photography" is a useful indicator to the relevant users. Fig. 7 further presents the comparison of pairwise attribute relation heatmaps. The left heatmap is the visualization of raw values of attribute relation computed using (1). The right one is the visualization of learned attribute relation compatibility using our model (3). We can see that there exist strong dependency correlations between attributes in the left figure while the learned attribute relations are relatively sparse yet clear. This is because that the raw values using (1) computed on the annotated dataset reflect the overall correlation statistics of the dataset. The raw values could be redundant and biased due to the size of dataset and annotation issue. Our relational SVM model incorporates the raw values of attribute relations to help learn attribute inference and attribute relation compatibility. The model encourages the helpful attribute pairs and suppresses the negative pairs. In this learning manner, the true attribute pairs will be maintained. Hence the learned attribute relations are accurate and useful for attribute inference. These results clearly demonstrate the effectiveness of the proposed model in mining popular attribute relation patterns.

2) User Attribute Inference: To train a model for each type of attribute, this type of attribute is selected as target attribute and

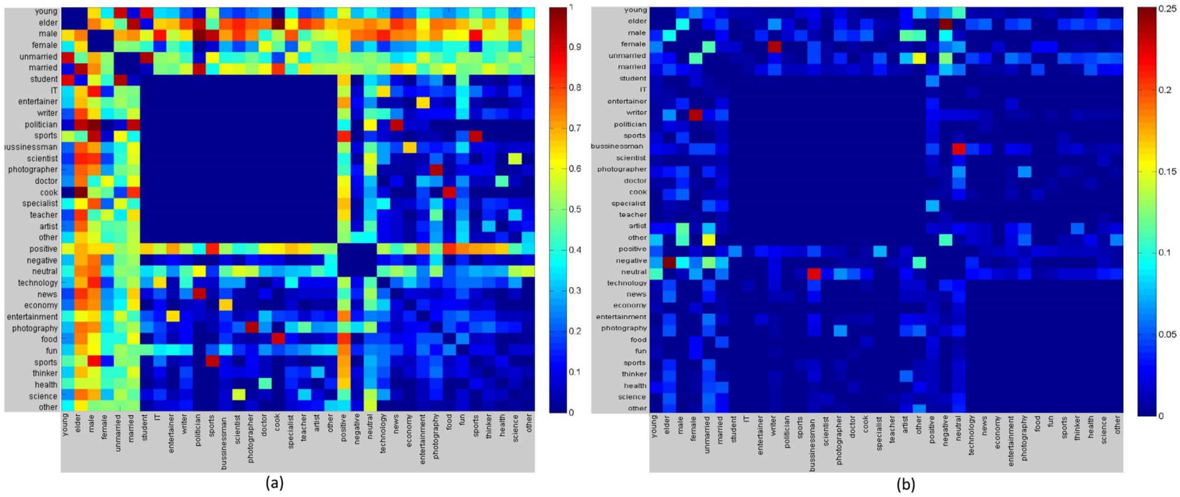

Fig. 7. Visualization of the attribute relation matrices. (a) The computed raw values of attribute relation with (1). (b) The inferred values of attribute relation using our model (3) (best viewed in color).

TABLE VI
PERFORMANCE COMPARISON OF DIFFERENT METHODS FOR USER ATTRIBUTE INFERENCE IN TERMS OF mAP

	Age	Gender	Relationship	Occupation	Interest	Emotional Orientation
SVM-Face	0.6194	0.7607	0.5835	0.0741	0.5005	0.3398
SVM-ProfilePhoto	0.5422	0.7185	0.5181	0.0776	0.5002	0.3579
SVM-PostPhoto	0.5047	0.6276	0.5193	0.1098	0.5215	0.3671
SVM-unigram	0.5989	0.7239	0.5899	0.2329	0.5490	0.4002
SVM-sociolinguistic	0.5972	0.7123	0.6081	0.2002	0.5501	0.3922
SVM-topic-based	0.5264	0.5768	0.5376	0.0798	0.5037	0.3333
Stacked SVM	0.6054	0.7856	0.6114	0.2373	0.5980	0.4096
Relational LSVM	**0.7278**	**0.7986**	**0.6240**	**0.2507**	**0.6172**	**0.4106**

the other types of attributes are treated as auxiliary attributes. Note that user interest attribute inference is a binary inference problem, where we train a model for each value of user interest. For instance, to train a model for value "technology", we take all user samples with the "technology" label to construct the positive set and the others to construct the negative set.

Table VI shows the performance of different methods for inferring user attributes, from which we can make the following observations. (1) Different types of user features contribute differently to user attribute inference. For example, the profile photo and face features are rather powerful for inferring user attribute "age" and "gender"; The term presence based SVM-unigram and SVM-sociolinguistic model performs better than other feature-based models in terms of "interest", "occupation" and "emotional orientation" attribute inference. This verifies the effectiveness of our discriminative term selection strategy. Concept-based image representation for post photos and LDA topic-based feature for text posts which show poor performance illustrates that extracting a compact and discriminative feature representation at user-level from user post content is very challenging. (2) Stacked SVM consistently outperforms SVMs based separately on the six user features in all user attributes inference cases. This confirms the superiority of stacked SVM in combing different types of user features for attribute inference. (3) The proposed Relational LSVM model incorporating user attribute relations significantly boosts the user attribute inference performance compared with the baseline methods. In particular, Relational LSVM improves the age attribute inference over stacked SVM by 12.24%. Other attributes inference performances are also improved. This result validates the advantage of exploiting attribute relations for user attribute inference.

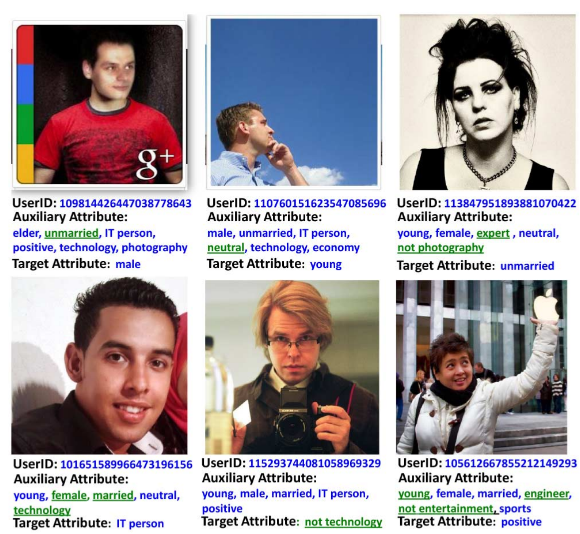

Fig. 8. Predicted user attributes of the testing sample users for user profiling. Failure instances are highlighted with green.

In Fig. 8, we present some attribute inference to construct the user profiles by the Relational LSVM model. We can see that the model accurately infers most of the user attributes, such as

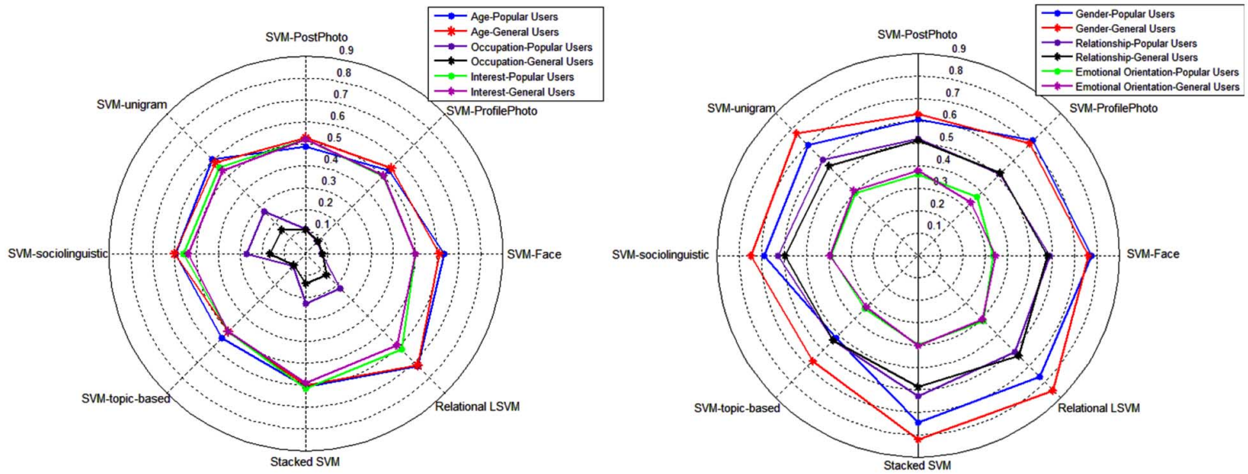

Fig. 9. Accuracies of user attribute inference on popular users and general users. Each polyline illustrates the attribute inference performance of different methods for one type of attributes on general users or popular users. See the legend for details.

target attribute "male", "young", "unmarried" and even auxiliary attribute "elder", "IT person", etc. However, we can also see that auxiliary attribute inference shows more incorrect instances than that of target attribute inference. This result is consistent with our discussions in Section V-D1.

3) Effect Analysis of Popular Users and General Users: We conduct an experiment to analyze the effect of user attribute inference on popular users and general users with the proposed relational LSVM model. Our collected dataset consists of popular users and general users. Popular users usually are well-known enough to have well-described Wikipedia pages. We use the Wikipedia API to determine whether a user is a popular user or a general user. We issue each user's name into Wikipedia API and determine the user's category. If a user is well described in a Wikipedia page, then we classify the user into the popular category, otherwise we classify the user into general category. This results in 1559 popular users and 989 general users. The user attribute inference methods are performed on general users and popular users respectively. Fig. 9 shows the result. From it, we can observe that there is no evident correlations between performance of attribute inference and users' category (general or popular). The attribute inference methods show slight better performance on popular users than that on general users in terms of attribute "age", "occupation", "interest", and "emotional orientation" while the methods achieve better results on general users for attribute "gender" and "relationship". These results are reasonable. The effectiveness of our relational LSVM model depends on user features and attribute relations extracted from user-generated multimedia content. Both popular users and general users with rich content could have good attribute inference results while the ones who have sparse user features are difficult to be predicted. Therefore, the utility of our model is irrelevant to that whether a user is a popular one or a general one. Table VII presents the ANOVA results of different attribute inference methods on popular users and general users. ANOVA is a widely used technique for the statistical test [2][11]. The ANOVA test reveals that the results of different inference methods are significant and confident.

TABLE VII
ANOVA RESULTS OF DIFFERENT METHODS ON POPULAR USERS AND GENERAL USERS

Methods	Popular Users			General Users		
	F	**df**	**Sig.**	**F**	**df**	**Sig.**
SVM-Face	31.66	5	<0.001	69.05	5	<0.001
SVM-ProfilePhoto	29.44	5	<0.001	15.7	5	<0.001
SVM-PostPhoto	17.94	5	<0.001	81.2	5	<0.001
SVM-unigram	43.03	5	<0.001	85.65	5	<0.001
SVM-sociolinguistic	15.13	5	<0.001	90.73	5	<0.001
SVM-topic-based	94.65	5	<0.001	86.42	5	<0.001
Stacked SVM	25.54	5	<0.001	66.34	5	<0.001
Relational LSVM	56.21	5	<0.001	10.53	5	<0.001

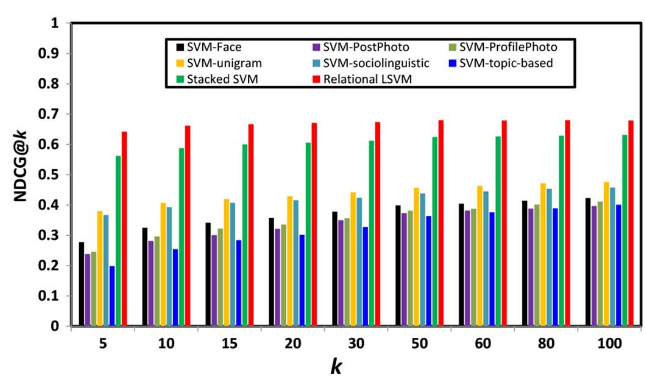

Fig. 10. Attribute-based user retrieval performance comparison.

4) Attribute-Based User Retrieval: Attribute-based user retrieval is to return the most related users when issuing an attribute-constructed query. This is different from user attribute inference which is to predict the most probable attribute labels given the input users. For evaluation, we create attribute-based structured queries from the annotated users on the dataset. We create three types of queries consisting of one attribute (denoted as Query1), two attributes (Query2), and three attributes (Query3) respectively. The total number of queries are 562, including 30 queries for Query1, 180 queries for Query2, and

[11]The value of F denotes the significance of the result, the larger F is, more significant the result is. df is short for degree of freedom. $Sig.$ denotes the probability we should deny the hypothesis. The smaller $Sig.$ is, more confident the result is.

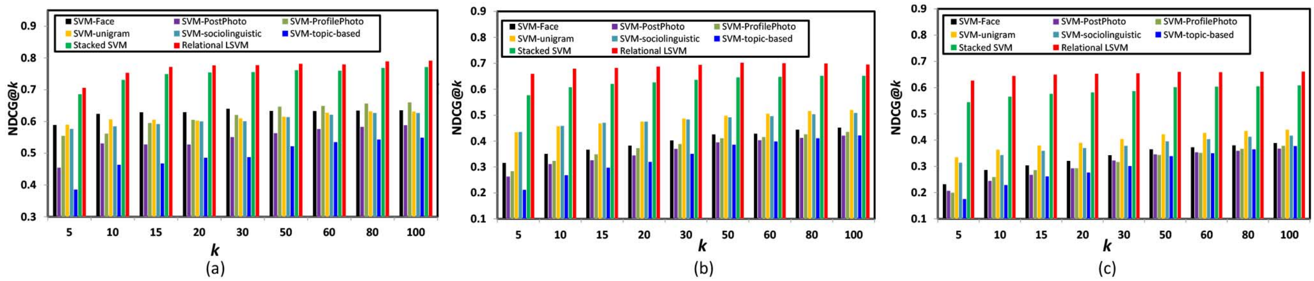

Fig. 11. Attribute-based user retrieval performance comparison for (a) Query1, (b) Query2, and (c) Query3.

352 queries for Query3. For Query1, we apply the model corresponding to the query attribute to score test users for ranking. For Query2 and Query3, each model corresponding to one attribute from the query is first used to score the test users. Then we use the sum of each attribute inference score to indicate the relevance of the testing users. For example, given a query of $\mathcal{Q} = \{$"elder, IT person, positive"$\}$, we first apply each classifier of the attribute in the query on the test users. Then, the response scores are combined and ranked for user retrieval. For our Relational LSVM model, we first select each attribute in the query as target attribute and other attributes as the auxiliary attributes. Equation (15) is then used to score the test users. We finally combine the response scores of each selected target attribute for ranking.

Fig. 10 shows the results of attribute-based user retrieval in terms of NDCG across all queries. It is clearly seen that the proposed Relational LSVM consistently and significantly outperforms the baseline methods, which is consistent with the results of user attribute inference. This validates the effectiveness of incorporating user attribute relations for the retrieval task. Fig. 11 further shows the retrieval performance comparison results for each type of query. In addition to the overall performance improvement of the Relational LSVM over the baseline methods, we can observe that the advantage of Relational LSVM is better reflected in more complex cases: Query2 in Fig. 11(b) and Query3 in Fig. 11(c). This indicates that utilizing the attribute relations is helpful for attribute-based user retrieval, especially for the complex queries consisting of multiple attributes.

Fig. 12 gives some exemplar attribute-based user retrieval results for the three types of queries by Relational LSVM model. Each row represents the retrieved results for one type of query. Returned users are ranked in descending order. The attributes concerned in the query are shown on the left side. Matched attributes in user profiles are highlighted in red, and incorrect samples are highlighted by green rectangles. We can see that most of the returned users match well with the queries. The first query is "photographer", and the top 5 returned users are all photographers. For the second query "female, unmarried", the failure user is male yet with the correct attribute unmarried. For the third query "elder, IT person, positive", we have Larry Page at the first rank and the failure user is young yet with the correct attribute IT person and positive. These results demonstrate the advantage of the proposed model for attribute-based user retrieval. Even for complex queries, the failure users are likely to have some matching attributes with the queries.

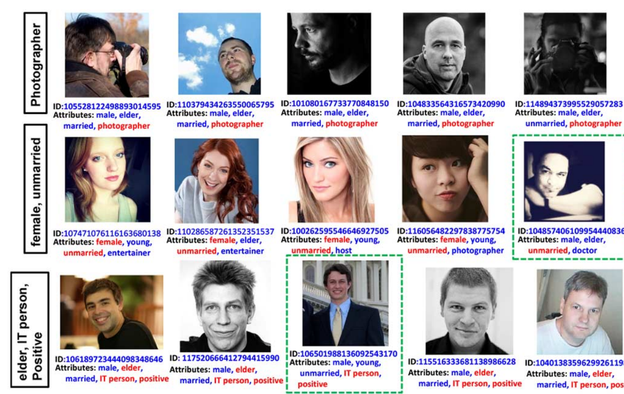

Fig. 12. Examples of attribute-based user retrieval for each type of query by the relational LSVM model.

C. Discussion

There are several important conclusions we can draw from the experimental results.

Relational user attribute inference: Inferring user attributes from user-generated multimedia information is an important and interesting problem. We systematically study how to infer different user attributes from online multimedia interaction such as profiles, posted texts and images. Our experimental results on Google+ set have justified that user attributes are predictable from their generated online multimedia content, which is consistent with the conclusion in previous work. In this work, we highlight the importance of attribute relations in attribute inference. The results in user attribute inference by our proposed relational LSVM model and attribute-based user retrieval have demonstrated the effectiveness of modeling the attribute relations. It is significantly important to emphasize that our proposed model for relational attribute inference is effective for popular users as well as general users with online generated content. The effectiveness of our model depends on user features and attribute relations extracted from user-generated multimedia content.

Performance analysis: We observed three major types of factors that affect the results. (1) *Predictable difficulty.* Our results reveal that it has different difficulty levels to infer different user attributes. It is more difficult to derive the multi-valued attributes (occupation, interest, emotional orientation) compared with binary valued attribute derivation (gender, age, and relationship). For example, most of user posts relate to some factual activities and do not express sentimental opinions, which makes it extremely difficult to judge the user emotional orientation. (2) *Data sparsity and missing issue.* Though we focus on the active users in Google+, there is still a portion of user's posts which are scarce or noisy. For example, some users only have reposts. For the users with scarce posts, we conduct a preprocessing step to remove such users. The user's posts are noisy. How to design effective user features for attribute inference is important. The results show that our textual and visual user features are effective for attribute inference. (3) *Unexpected annotation.* Though the available profiles from other platforms can help us ease the annotation, there still exist users with noisy posts and no referred sources to help annotation, which can make the attribute labeling unreliable and thus affect the evaluation. For example, it is difficult to accurately judge the occupation of some users due to the ambiguity of the posts to implicitly express the occupation information.

VII. Conclusion

In this paper, we have proposed a Relational LSVM model to address the problem of relational user attribute inference on user-generated multimedia information in social media. The extensive experiments have justified our motivation that exploring the dependency relations between attributes can help achieve better user attribute inference performance. The effectiveness of the whole framework is verified by combining the inferred attribute and mined attribute relation into the structured attribute-based user retrieval application.

In the future, we will investigate the following research directions: (1) The proposed framework is functional, with potential extensions to more practical applications, such as personalized recommendation, friend suggestion, etc. (2) Among the applications, attribute-based user retrieval is a most interesting one. We aim to design a ranking model specially towards the retrieval task. (3) A nonlinear version of the model, e.g., designing the potential functions with kernels, can be developed to improve the robustness and generalization capability.

References

[1] J. C. Aron Culotta and N. Ravi Kumar, "Predicting the demographics of Twitter users from website traffic data," in *Proc. 29th Conf. AI*, Jan. 2015, pp. 72–78.

[2] R. A. Bailey, *Design of Comparative Experiments*. Cambridge, U.K.: Cambridge Univ. Press, 2008, vol. 25.

[3] S. Bergsma, M. Dredze, B. V. Durme, T. Wilson, and D. Yarowsky, "Broadly improving user classification via communication-based name and location clustering on Twitter," in *Proc. Human Language Technol.: Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2013, pp. 1010–1019.

[4] S. Bergsma and B. V. Durme, "Using conceptual class attributes to characterize social media users," in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, 2013, pp. 710–720.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, no. 3, pp. 993–1022, 2003.

[6] C. Boulis and M. Ostendorf, "A quantitative analysis of lexical differences between genders in telephone conversations," in *Proc. 43rd Annu. Meeting Assoc. Comput. Linguistics*, 2005, pp. 435–442.

[7] J. D. Burger and J. C. Henderson, "An exploration of observable features related to blogger age," in *Proc. AAAI Spring Symp.: Comput. Approaches Analyzing Weblogs*, 2006, pp. 15–20.

[8] Z. Cheng, J. Caverlee, and K. Lee., "You are where you tweet: A content-based approach to geo-locating Twitter users," in *Proc. 9th ACM Int. Conf. Inf. Knowl. Manag.*, 2010, pp. 759–768.

[9] R. Cohen and D. Ruths, "Classifying political orientation on Twitter: It's not easy!," in *Proc. 7th Int. Conf. Weblogs Social Media*, 2013, pp. 91–99.

[10] M. Conover, B. Gonçalves, J. Ratkiewicz, A. Flammini, and F. Menczer, "Predicting the political alignment of Twitter users," in *Proc. IEEE 3rd Int. Conf. Privacy Security Risk Trust Social Comput.*, 2011, pp. 192–199.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2005, vol. 1, pp. 886–893.

[12] T. M. T. Do and T. Artières, "Large margin training for hidden Markov models with partially observed states," in *Proc. Int. Conf. Mach. Learn.*, 2009, p. 34.

[13] B. V. Durme, "Streaming analysis of discourse participants," in *Proc. Joint Conf. Empirical Methods Natural Language Process. Comput. Natural Language Learn.*, 2012, pp. 48–58.

[14] J. Eisenstein, B. O'Connor, N. A. Smith, and E. P. Xing, "A latent variable model for geographic lexical variation," in *Proc. Empirical Methods Natural Language Process.*, 2010, pp. 1277–1287.

[15] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *J. Mach. Learn. Res.*, vol. 9, pp. 1871–1874, 2008.

[16] P. F. Felzenszwalb, D. A. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2008, pp. 1–8.

[17] E. Filatova and J. M. Prager, "Occupation inference through detection and classification of biographical activities," *Data Knowl. Eng.*, vol. 76, pp. 39–57, 2012.

[18] N. Garera and D. Yarowsky, "Modeling latent biographic attributes in conversational genres," in *Proc. 47th Annu. Meet. Assoc. Comput. Linguistics 4th Int. Joint Conf. Natural Language Process.*, 2009, pp. 710–718.

[19] N. Garera and D. Yarowsky, "Structural, transitive and latent models for biographic fact extraction," in *Proc. 12th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2009, pp. 300–308.

[20] S. C. Herring and J. C. Paolillo, "Gender and genre variation in weblogs," *J. Sociolinguistics*, vol. 10, no. 4, pp. 439–459, 2006.

[21] G. Magno, G. Comarela, D. Sáez-Trumper, M. Cha, and V. Almeida, "New kid on the block: Exploring the Google+ social graph," in *Proc. Internet Meas. Conf.*, 2012, pp. 159–170.

[22] G. S. Mann and D. Yarowsky, "Multi-field information extraction and cross-document fusion," in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, 2005, pp. 483–490.

[23] A. Mislove, B. Viswanath, P. K. Gummadi, and P. Druschel, "You are who you know: Inferring user profiles in online social networks," in *Proc. 3rd ACM Int. Conf. Web Search Data Mining*, 2010, pp. 251–260.

[24] K. P. Murphy, Y. Weiss, and M. I. Jordan, "Loopy belief propagation for approximate inference: An empirical study," in *Proc. 15th Conf. Uncertainty AI*, 1999, pp. 467–475.

[25] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recog.*, vol. 29, no. 1, pp. 51–59, 1996.

[26] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in *Proc. Empirical Methods Natural Language Process.*, 2002, pp. 79–86.

[27] M. Pennacchiotti and A.-M. Popescu, "Democrats, republicans and Starbucks afficionados: User classification in twitter," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2011, pp. 430–438.

[28] D. Rao, D. Yarowsky, A. Shreevats, and M. Gupta, "Classifying latent user attributes in twitter," in *Proc. 2nd Int. Workshop Search Mining User-Generated Contents*, 2010, pp. 710–718.

[29] R. Sarawgi, K. Gajulapalli, and Y. Choi, "Gender attribution: Tracing stylometric evidence beyond topic and genre," in *Proc. 15th Conf. Comput. Natural Language Learning*, Portland, OR, USA, Jun. 2011, pp. 78–86.

[30] C. Tan, L. Lee, J. Tang, L. Jiang, M. Zhou, and P. Li, "User-level sentiment analysis incorporating social networks," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2011, pp. 1397–1405.

[31] M. Thomas, B. Pang, and L. Lee, "Get out the vote: Determining support or opposition from congressional floor-debate transcripts," in *Proc. Empirical Methods Natural Language Process.*, 2006, pp. 327–335.

[32] A. Torralba, K. P. Murphy, W. T. F. OjalaPH96, and M. A. Rubin, "Context-based vision system for place and object recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2003, vol. 1, pp. 273–280.

[33] S. Volkova and B. Van Durme, "Online Bayesian models for personal analytics in social media," in *Proc. 29th Conf. AI*, Jan. 2015, pp. 2325–2331.

[34] S. Volkova, T. Wilson, and D. Yarowsky, "Exploring demographic language variations to improve multilingual sentiment analysis in social media," *Proc. Empirical Methods Natural Language Process.*, pp. 1815–1827, 2013.

[35] J. Wang, J. Yang, K. Yu, F. Lv, T. S. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, Jun. 2010, pp. 3360–3367.

[36] I. Weber and C. Castillo, "The demographics of web search," in *Proc. 33rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2010, pp. 523–530.

[37] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, no. 5, pp. 241–259, 1992.

[38] C.-N. J. Yu and T. Joachims, "Learning structural SVMS with latent variables," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 1169–1176.

[39] K. Yu, G. Guan, and M. Zhou, "Resume information extraction with cascaded hybrid model," in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, 2005, pp. 499–506.

[40] F. A. Zamal, W. Liu, and D. Ruths, "Homophily and latent attribute inference: Inferring latent attributes of twitter users from neighbors," in *Proc. 6th Int. Conf. Weblogs Social Media*, 2012, pp. 387–390.

[41] E. Zheleva and L. Getoor, "To join or not to join: The illusion of privacy in social networks with mixed public and private user profiles," in *Proc. 18th Int. Conf. WWW*, 2009, pp. 531–540.

[42] L. Zhou, M. Ticrea, and E. H. Hovy, "Multi-document biography summarization," in *Proc. Empirical Methods Natural Language Process.*, 2004, pp. 434–441.

[43] J. Zhu, S. C. H. Hoi, M. R. Lyu, and S. Yan., "Near-duplicate keyframe retrieval by nonrigid image matching," in *Proc. ACM Multimedia*, 2008, pp. 41–50.

Quan Fang received the B.E. degree from Beihang University, Beijing, China, in 2010, and is currently working toward the Ph.D. degree at the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

His research interests include georeferenced social media mining and application, multimedia content analysis, knowledge mining, computer vision, and pattern recognition.

Mr. Fang was the recipient of the 2013 Microsoft Research Asia Fellowship. He received the Best Student Paper at Internet Multimedia Modeling 2013 and the Best Paper Finalist at ACM Multimedia 2013.

Jitao Sang received the B.E. degree from Southeast University, Nanjing, China, in 2007, and the Ph.D. degree from the Chinese Academy of Sciences (CASIA), Beijing, China, in 2012.

He is an Assistant Professor with the National Laboratory of Pattern Recognition, Institute of Automation, CASIA. He has authored or coauthored several refereed research papers. His research interests include multimedia content analysis, social media mining, and social network analysis.

Dr. Sang served as Special Session Organizer for MMM 2013 and Publication Chair for ACM ICIMCS 2013 and 2014. He was the recipient of the Special Prize of President Scholarship by the Chinese Academy of Sciences. He co-authored the Best Student Paper in Internet Multimedia Modeling 2013 and the Best Paper Candidate in ACM Multimedia 2012 and 2013.

Changsheng Xu (M'97–SM'99–F'14) is a Professor with the National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, and Executive Director of China-Singapore Institute of Digital Media, Singapore. He holds 30 granted/pending patents and has authored or coauthored over 200 refereed research papers. His research interests include multimedia content analysis/indexing/retrieval, pattern recognition, and computer vision.

Dr. Xu is an IAPR Fellow and ACM Distinguished Scientist. He is an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA, the *ACM Transactions on Multimedia Computing, Communications and Applications*, and the *ACM/Springer Multimedia Systems Journal*. He served as Program Chair of ACM Multimedia 2009. He has served as associate editor, guest editor, general chair, program chair, area/track chair, special session organizer, session chair, and TPC member for over 20 IEEE and ACM prestigious multimedia journals, conferences, and workshops. He was the recipient of the Best Associate Editor Award of the *ACM Transactions on Multimedia Computing, Communications and Applications* in 2012 and the Best Editorial Member Award of the *ACM/Springer Multimedia Systems Journal* in 2008.

M. Shamim Hossain (S'03–M'07–SM'09) received the Ph.D. degree in electrical and computer engineering from the University of Ottawa, Ottawa, ON, Canada.

He is an Associate Professor with King Saud University, Riyadh, Saudi Arabia. He has authored or coauthored more than 70 publications, including refereed IEEE/ACM/Springer/Elsevier journals, conference papers, books, and book chapters. His research interests include serious games, cloud and multimedia for health care, resource provisioning for big data processing on media clouds, and biologically-inspired approaches for multimedia and software system.

Dr. Hossain is a member of the ACM and ACM SIGMM. He has served as a member of the organizing and technical committees of several international conferences and workshops. He served as a Co-Chair of the 1st, 2nd, 3rd, 4th, and 5th IEEE ICME Workshop on Multimedia Services and Tools for E-Health. He served as a Co-Chair of the 1st Cloud-Based Multimedia Services and Tools for E-Health Workshop 2012 with ACM Multimedia. He currently serves as a Co-Chair of the 4th IEEE ICME Workshop on Multimedia Services and Tools for E-Health. He is on the editorial board of the *Springer International Journal of Multimedia Tools and Applications*. He was on the editorial board of some journals including the *International Journal of Advanced Media and Communication*. Previously, he served as a Guest Editor of the IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE and *Springer Multimedia Tools and Applications*. Currently, he serves as a Lead Guest Editor of the *Elsevier Future Generation Computer Systems*, *International Journal of Distributed Sensor Networks*, and *Springer Cluster Computing*.