Collective Semi-Supervised Learning for User Profiling in Social Media

Richard J. Oentaryo, Ee-Peng Lim, Freddy Chong Tat Chua, Jia-Wei Low, and David Lo

Abstract—The abundance of user-generated data in social media has incentivized the development of methods to infer the latent attributes of users, which are crucially useful for personalization, advertising and recommendation. However, the current user profiling approaches have limited success, due to the lack of a principled way to integrate different types of social relationships of a user, and the reliance on scarcely-available labeled data in building a prediction model. In this paper, we present a novel solution termed *Collective Semi-Supervised Learning* (CSL), which provides a principled means to integrate different types of social relationship and unlabeled data under a unified computational framework. The joint learning from multiple relationships and unlabeled data yields a computationally sound and accurate approach to model user attributes in social media. Extensive experiments using Twitter data have demonstrated the efficacy of our CSL approach in inferring user attributes such as account type and marital status. We also show how CSL can be used to determine important user features, and to make inference on a larger user population.

Index Terms—Convex optimization, collective learning, semi-supervised learning, social media, user profiling.

◆

1 INTRODUCTION

In recent years, we have witnessed a dramatic growth in social interactions taking place in social media such as Twitter and Facebook. These social media sites allow users to share contents (e.g., text, images, videos or web links), and to build social relationships, user communities, and common interest groups. Social media also generate a massive amount of digital data about user behaviors. The availability of such data has sparked the desire to learn more about consumers/users, fueling in turn the emergence of new services for peer interaction, marketing, and content sharing. For these services, there is a need to profile user preferences and attributes so as to support personalization, advertising, and recommendation [24], [28].

Despite the abundance of user-generated data, meta data about personal attributes that are directly useful for personalized services and recommendations are often not available. In Twitter, for instance, users rarely provide demographic information as gender, age, religion, or marital status. Such information can be used by Twitter or other organizations to perform market segmentation, contextualize search engine, or make better content/friend recommendations. Recent studies [18], [20], [24], [28] have nonetheless shown that it is possible to use statistical means to profile the latent user attributes, based on public data (e.g., users' contents and social ties) that the users reveal in social media.

Existing works on profiling latent user attributes in social media generally involve two types of data: content informa-

tion and social connectivity [14], [24], [28]. However, two major issues hinder the widespread use of these approaches. First, most (if not all) existing methods utilize users' own contents and/or one specific type of user relationship. They are not able to fuse different types of social relationship when one's own content is unavailable (e.g., a Twitter user who has no tweets), or when one type of relationship is not sufficiently informative of the attribute of interest. Second, the current approaches employ supervised learning methods that cannot generalize well when labeled data are scarce. This necessitates a more robust method that can also exploit a large pool of unlabeled data.

1.1 Motivating Example

To illustrate these more clearly, Figure 1 gives an example for the task of inferring if a Twitter user account is personal or organizational. The example consists of six personal and organization accounts whose labels are known (ovals and round boxes), and two accounts with unknown labels (dashed boxes). The upper half of Figure 1 shows two types of relationships: "follow" and "retweet from", and the tweet contents (words) of each user are shown in the left table. The bottom half shows the bag-of-words feature representation of the users. Traditionally, one can infer the label using only a user's own contents (self features). Often, however, the self features alone are not indicative of the label, e.g., for user *Andy* who never tweets. Augmenting social features derived from *Andy*'s followees (*Bob*, *Citibank* and *HSBC*) can provide stronger cues for Andy's label.

In this spirit, recent works [20], [21] have tried to incorporate social features, largely derived from a single type of relationship. But due to the sparse nature of users' connectivity, utilizing a single type of relationship may still be inadequate. For example, building a predictive model for *David*'s account type can hardly benefit from social ties if we consider only his "follow" connections, which he

- R. J. Oentaryo, E.-P. Lim, and D. Lo are with the Living Analytics Research Centre, Singapore Management University, Singapore 178902. E-mail: {roentaryo, eplim, davidlo}@smu.edu.sg

- F. C. T. Chua is with the Mechanism and Design Lab, HPLabs, Palo Alto, CA 94304, USA. E-mail: freddy.chua@hp.com

- J.-W. Low is with the Infocomm Development Authority, 10 Pasir Panjang Road, Singapore 117438. E-mail: davidlowjw@gmail.com

Fig. 1. Example of user profiling task

We summarize our main contributions as follows:

- We develop a simple method for collective learning via MRF, which takes into account—for different types of relationships—both the tie strength between a given user and its neighbors as well as the features of the neighbors. Deviating from existing multi-relational learning methods, which either treat multi-relational information as constraints to the learning process [32], [33] or rely on low-rank assumption to decompose multi-relational data [7], [13], [29], our MRF approach is more general and makes less restrictive assumption about the multi-relational information.

- We put forward the concept of CD regularization that offers a convex formulation of semi-supervised learning using unlabeled data instances (i.e., unlabeled users). This leads to a computationally sound learning procedure that warrants a unique, globally optimal solution, which can be readily identified via off-the-shelf numerical optimization methods (e.g., the Quasi-Newton algorithm in [22]). This makes our CSL approach accurate, robust, and scalable.

- We extensively evaluate our CSL approach through two user profiling tasks in Twitter: modeling users' *account type* and *marital status*. The results demonstrate the accuracy and robustness of our approach, and how different relationship types and unlabeled data contribute to its performance. We also show how CSL can be used to unravel important features for different relationship types, and to infer/profile on a larger user population in Twitter.
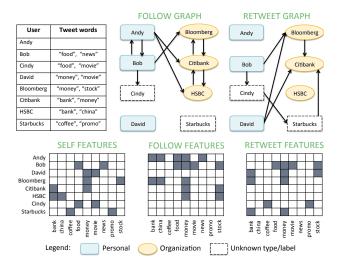
has none. Hence, we are not able to gain additional word features from his followers. Similarly for *Andy*, *Citibank* and *HSBC*, there is no additional word feature if we use only the "retweet" links, i.e., all of them do not have "retweet" links. Intuitively, integrating social features from multiple, complementary sources can boost the confidence in the label prediction. For instance, modeling *Bob*'s label can benefit from the co-occurrence of the words that appear in his follow and retweet features. Finally, it is possible to build a more robust model by utilizing unlabeled data. For instance, *Bob*'s shares a common word "food" with *Cindy*'s, but has no common word with *Starbucks*. Exploiting this, we can create a better classification boundary that makes *Bob*'s label closer to *Cindy*'s' and further away from *Starbucks*.

1.2 Proposal and Contributions

Generalizing the above scenario, we propose a new take on user profiling task by answering several research questions:

- How can we exploit multiple types of social relationship and unlabeled data in order to infer/profile the (latent) user attributes better?
- How do we develop an efficient and robust profiling method that can integrate multiple relationship types and unlabeled data in a computationally sound way?
- Can we understand the contributions of different features and relationship types, as well as infer/profile on a larger user population?

In light of these questions, we present in this paper a new computational method dubbed *collective semi-supervised learning* (CSL), for modeling user attributes in social media. To our best knowledge, this work is the first to formulate user profiling as the problem of jointly exploiting multiple relationship types and unlabeled data, and hence CSL provides a unified approach to solve this problem in a computationally principled and efficient manner. In particular, CSL models multiple relationship types by generically constructing *multi-relational features* (MRF), and then integrates the concept of *convex divergence* (CD) regularization in order to establish a convex formulation of semi-supervised learning utilizing unlabeled data.

1.3 Paper Outline

The remainder of this paper is organized as follows. In Section 2, we first give an overview of related works. Section 3 elaborates the proposed CSL approach. In Section 4, we describe the user profiling tasks addressed in this work, followed by the corresponding experimental results and analyses in Section 5. Finally, we conclude in Section 6.

2 RELATED WORK

We first survey related works on user attribute profiling, semi-supervised learning, and multi-relational learning. We then discuss how our approach differs from these works.

2.1 User Attribute Profiling

Several works have recently been developed to infer multiple user attributes in social media. Rao *et al.* [28] proposed a set of network structure-based features to infer the attributes of Twitter users, including gender, age, geographical origin, and political preference. These features were then fed into a stacked classifier to infer the attributes. Mislove *et al.* [24] used both global and local community detection methods in order to find communities of users who share common attribute values. Ikeda *et al.* [14] used social communities to infer the demographic information of the Twitter users. They developed a hybrid method that uses text features as well as network structure features.

Recently, Kosinski *et al.* [18] showed that public information obtained from Facebook can be used to predict demographic attributes of users. By factorizing a sparse matrix representing which user likes which subject/topic, a low-rank representation of the attributes was obtained and then used as features for regression and classification. Li *et al.* [20] devised a distant supervised learning method to infer the attributes of Twitter users by augmenting structured auxiliary data from the Facebook and Google+ networks. The unstructured information is matched with the structured ground truth to increase the prediction accuracy.

In [21], Li *et al.* developed a new co-profiling to jointly infer the users' attributes (partially observed) and relationship type (completely unobserved) within the users' ego network (observed). The underlying assumption is that social connections are discriminatively correlated with user attributes (e.g., employer, college) via a hidden relationship type (e.g., colleague, classmate). Dong *et al.* [8] presented a factor graph model to predict the demographic attributes of mobile phone users. The model utilizes three types of factor: attribute factor, dyadic factor, and triadic factor, which represent correlation between the user's attributes and his/her network characteristics, between the attributes of two users, and among the attributes of user triads, respectively.

2.2 Semi-Supervised Learning

The literature on SSL methods is vast, so here we review only methods that are most related to our work. The simplest form of SSL is *bootstrapping*, whereby a classifier is first trained using labeled data, and then applied to unlabeled data so as to generate more labeled samples for the next rounds of training [1], [12]. Bootstrapping works based on a simplistic assumption is that the classifier's own (high-confidence) predictions are correct. *Co-training* is an extension of bootstrapping in which two (or more) classifiers are trained on different, ideally disjoint sets of features, and generate labeled samples to improve each other [5], [25]. This method is less prone to mistakes than bootstrapping, but it requires that natural feature splits exist in the data.

Another class of SSL methods uses the *low-density assumption* [6], encouraging the decision boundary to lie in low-density regions for improving generalization results. The most common way to achieve this is to use a maximum margin algorithm such as transductive support vector machine [17]. However, the corresponding learning problem is nonconvex, which is hard to solve and does not warrant globally optimal solution. Grandvalet and Bengio [11] devised an alternative method based on *entropy regularization* (ER). This approach encourages the posterior probability to be closer to 1 or 0 through any high-density region, while the decision boundary corresponds to intermediate probability. Again, however, the resulting problem is nonconvex.

There are also active research works on *graph-based* SSL (GSSL) methods, which treat both labeled and unlabeled data as nodes in a graph and build edges between pairs of nodes weighted by their affinities (similarities) [6]. A popular example of GSSL methods is label spreading [36], which iteratively propagates a node's label distribution to its neighbors according to their affinity. Generalizations have been proposed under the umbrella of manifold regularization [4], [6]. We note, however, that the GSSL methods

work well only when the affinity or manifold assumption holds for the data, i.e., nodes that are similar would likely have similar label distribution. As such, the GSSL methods requires the right choice of affinity graph to work well.

Extending the entropy regularization method [11], several *information-theoretic* SSL methods have been developed [23], [27]. Mann and McCallum [23] proposed the expectation regularization (XR) to build a simple and robust SSL method. The XR augments the learning procedure with a regularization term that minimizes the Kullback-Leibler (KL) divergence between the label expectations predicted by the model and human-provided label expectation priors. More recently, Niu *et al.* [27] devised a squared-loss mutual information regularization (SMIR) method, which led to a convex SSL problem formulation guaranteed under a mild condition. The key appeal of this approach is that an analytical (closed-form) solution can be computed to identify unique, globally optimal model parameters [27], [31].

2.3 Multi-Relational Learning

Multi-relational learning (MRL) is applicable when the data are available in multiple structured formats and can be represented as multiple graphs (a.k.a. *multigraph*). That is, a multigraph can be used in MRL to encode different types of relationship (edge) among entities (nodes). In [35], Xu *et al.* presented a seminal work on multi-relational Gaussian process (MRGP) that utilizes a generative probabilistic model based on Gaussian process. It combines the covariance and random variables approaches to model multiple relations, which in turn provides support for multiple relational learning tasks with multiple types of entities and relations.

In a different task domain, Wang *et al.* [33] proposed a MRL method for video annotation that integrates multiple graphs into a regularization framework, so as to sufficiently exploit their complementation. This method was shown to be equivalent to first fusing multiple graphs and then conducting graph-based SSL on the fused graph. A similar approach was used in [32] to tackle the task of protein domain ranking in structural biology. In this approach, the intrinsic manifold of protein domain distribution was approximated by combining multiple graphs for regularization.

Another branch of relational learning considers the relations among entities as resulting from the latent factors of these entities. These approaches often translate into learning an embedding of the entities, which corresponds to a *matrix factorization* problem. This can be naturally extended to MRL by stacking the matrices to be factorized and then applying tensor factorization methods [7], [13]. Another natural extension to MRL is to share the common embedding or the entities across relations via collective matrix factorization [26], [29]. This method has shown state-of-the-art performances on relational datasets [26], although the number of relation types is usually modest (less than 100). Extensions have recently been proposed in [9], [15] to handle multi-relational data with a large number of relation types.

2.4 Our Approach

Our CSL approach differs from the existing works in several important ways, which we elaborate below.

Comparisons with existing user profiling methods. While many of the current profiling methods utilize users' social information for attribute predictions, they have focused on just one type of relationship (e.g., only the follow relationship in Twitter), lacking a systematic method to incorporate different types of relationships altogether. Second, the existing profiling methods utilize only labeled data, which are often very scarce. A more robust predictive model can be obtained by also exploiting a large pool of unlabeled data. CSL offers these two capabilities in a unified and synergistic manner, which—to the best of our knowledge—is the first of its kind for user profiling applications.

Comparisons with existing SSL methods. In contrast to conventional SSL methods such as bootstrapping [1] and co-training [5], our CSL approach does not rely on the assumption that the model's own (high-confidence) prediction is correct, or that natural feature splits exist in the data. Compared to the GSSL methods such as label spreading [36]—whose performance is sensitive to the choice of affinity graph—our approach works based on the empirical distribution of unlabeled data, which is simpler and less restrictive. Our approach also provides a convex formulation of SSL that is more robust and computationally elegant than the ER method [11], whereby the learning procedure can be easily trapped to one of the (multiple) local optimal solutions. Finally, the CSL approach is more general than the state-of-the-art information-theoretic SSL methods such as XR [23] and SMIR [27]. These methods have not accounted for multi-relational information in their formulation.

Comparisons with existing MRL methods. Our CSL approach compares favourably to the MRGP method [33] in several ways. First, CSL adopts a discriminative probabilistic model, which should in principle be more accurate than the generative model used in MRGP [30]. Second, MRGP handles only binary relations (graphs), whereas ours can take weighted graphs. Third, MRGP is trained using the expectation maximization (EM) algorithm, which does not warrant a globally optimal solution. CSL is also less restrictive than the MRL methods in [32], [33]. The latter treat multi-relational information (i.e., multigraph) as constraints to the learning process, whereas ours casts multi-relational information into multi-relational features that in turn serve as additional information to be augmented into the learning process. Finally, CSL is more generic/flexible than the matrix/tensor factorization methods [7], [9], [13], [15], [26], [29]. These methods rely on low-rank assumption for matrix/tensor decomposition, and do not yet cater for explicit (i.e., non-latent) features defined for each entity.

3 PROPOSED FRAMEWORK

Our CSL framework operates based on two inputs: 1) partially labeled data, comprising a feature matrix $\mathbf{X}$ with known labels $\mathbf{Y}_L$ and missing labels $\mathbf{Y}_U$, and 2) multigraph, composed of multiple directed graphs $\mathbf{G}_m$ that encode different types of social relationship. We first describe our notations: Let $\mathbf{G} = \{\mathbf{G}_1, \ldots, \mathbf{G}_m, \ldots, \mathbf{G}_M\}$ be a multigraph composed of M graphs, where each graph $\mathbf{G}_m = (\mathbf{V}, \mathbf{E}_m)$ comprises nodes $\mathbf{V}$ and edges $\mathbf{E}_m$. Note that here we have a common set of nodes $\mathbf{V}$, but different sets of edges $\mathbf{E}_m$. We denote the feature matrix of nodes $\mathbf{V}$ as $\mathbf{X} \in \mathbb{R}^{N \times J}$, and the weighted adjacency matrix of edges $\mathbf{E}_m$ as $\mathbf{W}_m \in \mathbb{R}^{N \times N}$, where N and J are the number of nodes and features respectively. We also denote the labels of $\mathbf{V}$ as $\mathbf{Y}$, representing the user attributes of interest. Lastly, a node maps exactly to a data instance, so we shall use the two terms interchangeably.

3.1 Probabilistic Foundation

We first outline the probabilistic formulation of our CSL approach here. Ultimately, our goal is to maximize the posterior distribution of the model parameters Θ, given the labels $\mathbf{Y}$, node features $\mathbf{X}$, and multigraph $\mathbf{G}$. Here the posterior can be computed using the Bayes' rule:

$$p(\Theta|\mathbf{Y},\mathbf{X},\mathbf{G}) = \frac{p(\Theta,\mathbf{Y},\mathbf{X},\mathbf{G})}{p(\mathbf{Y},\mathbf{X},\mathbf{G})} = \frac{p(\mathbf{Y}|\mathbf{X},\mathbf{G},\Theta)p(\Theta)}{p(\mathbf{Y},\mathbf{X},\mathbf{G})}$$
$$\propto p(\mathbf{Y}|\mathbf{X},\mathbf{G},\Theta)p(\Theta) \tag{1}$$

In this work, we focus on *partially labeled* data, whereby only a few data instances have observed labels $\mathbf{Y}_L$, while the remaining instances are largely unlabeled, i.e., their labels $\mathbf{Y}_U$ are assumed to be *missing at random* [11]. Since $\mathbf{Y} = \mathbf{Y}_L \cup \mathbf{Y}_U$, it follows that $p(\mathbf{Y}|\mathbf{X},\mathbf{G},\Theta) = p(\mathbf{Y}_L|\mathbf{X},\mathbf{G},\Theta)p(\mathbf{Y}_U|\mathbf{X},\mathbf{G},\Theta)$ and the posterior becomes:

$$p(\Theta|\mathbf{Y},\mathbf{X},\mathbf{G}) \propto p(\mathbf{Y}_L|\mathbf{X},\mathbf{G},\Theta)p(\mathbf{Y}_U|\mathbf{X},\mathbf{G},\Theta)p(\Theta) \tag{2}$$

where $\mathbf{Y}_L$ and $\mathbf{Y}_U$ are treated as conditionally independent. In turn, we can maximize the posterior $p(\Theta|\mathbf{Y},\mathbf{X},\mathbf{G})$ by minimizing its negative logarithm (a.k.a. *loss function*) $\mathcal{L}$:

$$\mathcal{L} = -\ln(p(\mathbf{Y}_L|\mathbf{X},\mathbf{G},\Theta)) - \ln(p(\mathbf{Y}_U|\mathbf{X},\mathbf{G},\Theta)) - \ln(p(\Theta)) \tag{3}$$

For convenience, we break (3) into two parts, respectively:

$$\mathcal{L}_L = -\ln(p(\mathbf{Y}_L|\mathbf{X},\mathbf{G},\Theta)) - \ln(p(\Theta)) \tag{4}$$
$$\mathcal{L}_U = -\ln(p(\mathbf{Y}_U|\mathbf{X},\mathbf{G},\Theta)) \tag{5}$$

Remark. It must be noted that the above formulation is new and different from the contemporary user attribute profiling methods [8], [14], [20], [21], [24], [28]. All these approaches focus only on a single type of relationship, whereas ours can readily cater for multiple types of relationship (i.e., multigraph $\mathbf{G}$). Moreover, the existing methods do not yet exploit the additional information provided by unlabeled data (i.e., $\mathbf{Y}_U$) in guiding their learning processes.

3.2 Base Model

The proposed CSL approach can be viewed as a generalization of the contemporary *logistic regression* model [10]. Traditionally, logistic regression learns in a fully supervised fashion based solely on the labeled data $\mathbf{Y}_L$ (i.e., it does not use $\mathbf{Y}_U$), and it does not take into account multi-relational information encoded as multigraph $\mathbf{G}$. That is, by excluding $\mathbf{Y}_U$ and $\mathbf{G}$ and by assuming independent and identically distributed (i.i.d) data instances, logistic regression essentially learns to minimize the following loss function:

$$\mathcal{L}_L = -\sum_{i=1}^{L} \ln(p(y_i|x_i,\Theta)) - \sum_{j=1}^{J} \ln(p(\theta_j)) \tag{6}$$

where $y_i \in \mathbf{Y}_L$ and $x_i \in \mathbf{X}$ are the actual label and feature vector for data instance i respectively, $L = |\mathbf{Y}_L|$ is the

number of labeled data instances, and $\theta_j \in \Theta$ is a model (i.e., weight) parameter that we want to learn for each feature j.

Without loss of generality, we consider binary class label[1] $y_i \in \{0, 1\}$. For binary classification, we may take that each sample likelihood $p(y_i|x_i, \Theta)$ follows a *Bernouli* distribution (which is analogous to the toss of a coin):

$$p(y_i|x_i, \Theta) = \sigma_i^{y_i}(1 - \sigma_i)^{(1-y_i)} \tag{7}$$

where $\sigma_i = \sigma(f(x_i, \Theta)) = \frac{1}{1+\exp(-f(x_i,\Theta))}$ refers to the logistic function, and $f(x_i, \Theta)$ is the linear model:

$$f(x_i, \Theta) = \sum_{j=1}^{J} \theta_j x_{i,j} \tag{8}$$

For the prior $p(\theta_j)$, we use a Gaussian distribution with zero mean and inverse variance λ:

$$p(\theta_j) = \frac{1}{Z} \exp\left(-\frac{\lambda}{2}\theta_j^2\right) \tag{9}$$

where Z is a normalizing constant and $\lambda > 0$. Accordingly, we can write the overall loss $\mathcal{L}_L$ for logistic regression as:

$$\mathcal{L}_L = -\sum_{i=1}^{L}[y_i \ln(\sigma_i) + (1 - y_i)\ln(1 - \sigma_i)] + \frac{\lambda}{2}\sum_{j=1}^{J}\theta_j^2 \tag{10}$$

Note that the regularization term $\frac{\lambda}{2}\sum_{j=1}^{J}\theta_j^2$ serves to penalize large values of the model parameters θ_j, thereby reducing the risk of data overfitting [10].

3.3 Multi-Relational Features

We now extend the base logistic regression model to incorporate the multi-relational information $\mathbf{G}$ through adding multi-relational features (MRF). Specifically, by incorporating $\mathbf{G}$ into the parameterization of the base model in (8), we obtain an extended linear model $f(x_i, \mathbf{G}, \Theta)$:

$$f(x_i, \mathbf{G}, \Theta) = \sum_{j=1}^{J}\theta_j x_{i,j} + \sum_{m=1}^{M}\sum_{j=1}^{J}\alpha_{m,j}\pi_{m,i,j} \tag{11}$$

and correspondingly $\sigma_i = \sigma(f(x_i, \mathbf{G}, \Theta))$, where $\pi_{m,i,j}$ is the jth *relational feature* of data instance (i.e., node) i for graph $\mathbf{G}_m$, and $\alpha_{m,j}$ is the corresponding j^{th} *relational weight* for $\mathbf{G}_m$, and $\Theta = \{\theta_j\} \cup \{\alpha_{m,j}\}$ is the set of all model parameters. Under this notation, we call θ_j as the *self weight* corresponding to the *self features* $x_{i,j}$ of node i.

There are numerous ways to define the relational feature $\pi_{m,i,j}$ of a node i. In principle, one can derive the relational features through an arbitrary *aggregation* function summarizing some global or local properties of each graph $\mathbf{G}_m$, and the aggregation function need not be the same for different graphs $\mathbf{G}_m$. For efficiency and interpretability, however, in this work we focus on a simple aggregation function that combines the information from only the *immediate neighbors* of a node i by taking a weighted average of their features:

$$\pi_{m,i,j} = \frac{\sum_{(i,i') \in \mathbf{E}_m} w_{m,i,i'} x_{i',j}}{\sum_{(i,i') \in \mathbf{E}_m} w_{m,i,i'}} \tag{12}$$

1. Extension to multi-class task with $C > 2$ labels is straightforward, which can be done by constructing C binary logistic regression models.

where $w_{m,i,i'} \in \mathbf{W}_m$ represents the tie strength of a node (instance) i with its neighbor i' in graph $\mathbf{G}_m$. For instance, in the context of Twitter follow graph, the notion of neighbors refers to the followees of a given user.

With the addition of the relational weights $\alpha_{m,j}$, the penalized loss $\mathcal{L}_L$ now becomes:

$$\mathcal{L}_L = -\sum_{i=1}^{L}[y_i \ln(\sigma_i) + (1 - y_i)\ln(1 - \sigma_i)] + \frac{\lambda}{2}\sum_{j=1}^{J}\left(\theta_j^2 + \sum_{m=1}^{M}\alpha_{m,j}^2\right) \tag{13}$$

Remark. The MRF formulation in (11) and (12) provides a simple yet powerful way to incorporate multiple types of social relationship into user attribute prediction. Such formulation has several key appeals:

- Unlike previous MRL methods that use multigraph to *constrain* the learning processes, e.g., [32], [33], or rely only on *latent features*, e.g., [7], [13], [26], our MRF formulation is more generic and makes less stringent assumption. That is, we treat multigraph as *additional source* of information, and we can use any aggregation function to summarize this information.
- Our MRF formulation can also readily cater for different types of features, such as numeric features (e.g., tweet count), n-gram representation of text features, or binary vector of categorical features. For ease of interpretation/analysis, though, we shall focus on the n-gram text features in this work.
- By aggregating and augmenting the neighbors' features on a per-graph basis, we can exploit the dependencies and complementarity among various features, while preserving the semantics of each type of relationship. Especially, the learned relational weights $\alpha_{m,j}$ can be used to understand the contribution and importance of different types of relationship in modeling latent user attributes.
- From a computational standpoint, the MRF formulation is efficient. First, the aggregation function keeps the problem dimensionality moderate; we only require $(M+1) \times J$ features instead of naïvely appending all neighbors' features. Second, the relational features $\pi_{m,i,j}$ can be pre-computed once for every instance/node i prior to parameter learning process. Finally, our MRF formulation maintains the linearity of our model (11), which preserves the convexity of the overall loss (3) (see Section 3.5).

It is also worth noting that our MRF formulation is different from that of conditional random field (CRF) [30]. The CRF approach usually involves some form of dependencies among the (output) labels y_i, whereas our MRF method focuses on the dependencies in the input space x_i and assumes that the labels y_i are (conditionally) independent. While structured modeling via CRF can potentially improve performance, it comes at the expense of higher computational complexity and degraded model interpretability. As such, we do not pursue the CRF approach in this work.

3.4 Convex Divergence Regularization

After constructing the MRF for all data instances (both labeled and unlabeled), CSL carries out a semi-supervised learning (SSL) using unlabeled data for improving model generalization and robustness. To this end, we put forward the idea of *convex divergence* (CD) to regularize learning via unlabeled data. The CD regularization stems from the following definition of $p(\mathbf{Y}_U|\mathbf{X}, \mathbf{G}, \Theta)$ in (2):

$$p(\mathbf{Y}_U|\mathbf{X}, \mathbf{G}, \Theta) = \frac{1}{Z} \exp\left(-\beta D_f(\mu||\rho)\right) \qquad (14)$$

where Z is a normalizing constant, β is a (nonnegative) user-specified regularization parameter, and $D_f(\mu||\rho)$ is the f-divergence [2] between two distributions μ and ρ:

$$D_f(\mu||\rho) = \sum_z f\left(\frac{\mu(z)}{\rho(z)}\right)\rho(z) \qquad (15)$$

which is defined over some space z and $f : [0, +\infty) \to \mathbb{R}^+$ is a continuous convex function, such that $f(1) = 0$.

We note here that $D_f(\mu||\rho)$ does not uniquely define the form of the prior distribution $p(\mathbf{Y}_U|\mathbf{X}, \mathbf{G}, \Theta)$, but the latter can be constructed through constraints imposed by $D_f(\mu||\rho)$. Also, to ensure convexity in the overall loss $\mathcal{L}$, it is necessary to choose the distributions μ and ρ such that $D_f(\mu||\rho)$—or its approximation—is twice-differentiable, and its second derivative $\bigtriangledown^2 D_f(\mu||\rho)$ is non-negative for all possible values of z.

In this work, we focus on an instantiation of $D_f(\mu||\rho)$ that involves computing the *Kullback-Leibler* (KL) divergence between some class prior $\tilde{p}$ and the expected predictions $\mathbb{E}[\sigma_i]$ made by the model on unlabeled data:

$$
\begin{aligned}
D_{KL}(\mu||\rho) &= \tilde{p}\ln\left(\frac{\tilde{p}}{\mathbb{E}[\sigma_i]}\right) \\
&= -\tilde{p}\ln(\mathbb{E}[\sigma_i]) + \underbrace{\tilde{p}\ln(\tilde{p})}_{=constant} \\
&\propto -\tilde{p}\ln(\mathbb{E}[\sigma_i]) \qquad (16)
\end{aligned}
$$

where the function f is defined as $f(t) = t\ln(t)$. In this case, $D_{KL}(\mu||\rho) = 0$ when μ and ρ match exactly. Our goal here is to *minimize* (16), implying that we want to obtain a classification model such that the expectation of its predictions on unlabeled data is similar to the class prior.

The class prior can be either (manually) specified based on domain knowledge, or computed based on the class distribution on the labeled data. For simplicity, we choose the latter approach in this work, by defining $\tilde{p}$ as $\tilde{p} = \frac{1}{L}\sum_{i=1}^L y_i$, i.e., the proportion of positive instances in the labeled data.

Finally, by substituting (16) into (14) and dropping constant terms, we obtain the CD regularization:

$$
\begin{aligned}
\mathcal{L}_U &= -\ln\left(p(\mathbf{Y}_U|\mathbf{X}, \mathbf{G}, \Theta)\right) \\
&\propto -\ln\left(-\beta D_{KL}(\tilde{p}||\hat{p}_\Theta)\right) \\
&\propto -\beta\,\tilde{p}\ln\left(\mathbb{E}[\sigma_i]\right) \qquad (17)
\end{aligned}
$$

Note that (17) is not convex in its current form. Fortunately, we can use the *Jensen inequality* [16] in order to derive a convex upper bound of (17). The Jensen inequality states that the expectation of a convex function φ is equal to or greater than the function of the expectation, i.e., $\mathbb{E}[\varphi(x)] \geq \varphi(\mathbb{E}[x])$. It then follows that the convex upper bound is:

$$
\begin{aligned}
\mathcal{L}_U &\propto \beta\,\tilde{p}\left(-\ln\left(\mathbb{E}[\sigma_i]\right)\right) \\
&\leq \beta\,\tilde{p}\,\mathbb{E}\left[-\ln\left(\sigma_i\right)\right] \qquad (18)
\end{aligned}
$$

with $\varphi(x) = -\ln(x)$. Subsequently, we can approximate the expectation via an empirical average $\mathbb{E}\left[-\ln\left(\sigma_i\right)\right] = -\frac{1}{U}\sum_{i=L+1}^{L+U}\ln(\sigma_i)$, where $U = |\mathbf{Y}_U|$ is the total number of unlabeled data instances. This leads to a new convex formulation of $\mathcal{L}_U$ using unlabeled data:

$$\mathcal{L}_U \propto -\beta\,\tilde{p}\sum_{i=L+1}^{L+U}\ln(\sigma_i) \qquad (19)$$

whereby we absorb the term $\frac{1}{U}$ into β for simplicity.

Remark. The formulation in (19) is related to the XR approach [23], with some key differences. First, the XR method tries to minimize (16) directly, which is non-convex and may lead to local optima. In contrast, our CD formulation aims at reducing (16) by minimizing its convex upper bound (19), which is simpler and computationally more appealing (due to convexity). Second, our formulation generalizes XR by not only learning from unlabeled data, but also taking into account the different types of relationship among instances via MRF. We will empirically show in Section 5 how multi-relational information and unlabeled data can work together to improve user profiling performances. Our formulation is also conceptually superior to that of the SMIR method [27], whose convexity is not guaranteed when the L2 regularization parameter (λ in our notation) is not sufficiently large [27]. We will show in Section 3.5 that our CSL formulation imposes strict convexity for any positive λ.

3.5 Parameter Learning

We can now combine (17), (13) and (11) to derive the overall loss function $\mathcal{L}$ for the CSL approach, which is given by:

$$
\begin{aligned}
\mathcal{L} = &-\sum_{i=1}^L \left[y_i\ln(\sigma_i) + (1-y_i)\ln(1-\sigma_i)\right] \\
&+ \frac{\lambda}{2}\sum_{j=1}^J\left(\theta_j^2 + \sum_{m=1}^M \alpha_{m,j}^2\right) - \beta\,\tilde{p}\sum_{i=L+1}^{L+U}\ln(\sigma_i) \quad (20)
\end{aligned}
$$

with $\sigma_{i,1} = \sigma(f(x_i, \mathbf{G}, \Theta))$ and $\sigma_{i,2} = 1 - \sigma(f(x_i, \mathbf{G}, \Theta))$.

Convexity. Before we proceed with the learning procedure for minimizing $\mathcal{L}$, we first give a proof sketch for the convexity of $\mathcal{L}$. This is done by examining the *slope* (i.e., first derivative) and *curvature* (i.e., second derivative) of $\mathcal{L}$. Firstly, the slope with respect to each self weight θ_j is:

$$\frac{\partial \mathcal{L}}{\partial \theta_j} = -\sum_{i=1}^L \frac{y_i}{\sigma_i}\frac{\partial \sigma_i}{\partial \theta_j} + \lambda\theta_j - \beta\,\tilde{p}\sum_{i=L+1}^{L+U}\frac{1}{\sigma_i}\frac{\partial \sigma_i}{\partial \theta_j} \qquad (21)$$

and it is easy to show that $\frac{\partial \sigma_i}{\partial \theta_j} = \sigma_i(1-\sigma_i)x_{i,j}$, where $\sigma_i = \sigma(f(x_i, \mathbf{G}, \Theta))$. We can then evaluate the slope as:

$$
\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \theta_j} = &-\sum_{i=1}^L\left[\left(\frac{y_i}{\sigma_i} - \frac{1-y_i}{1-\sigma_i}\right)\frac{\partial \sigma_i}{\partial \theta_j}\right] + \lambda\theta_j - \beta\,\tilde{p}\sum_{i=L+1}^{L+U}\frac{1}{\sigma_i}\frac{\partial \sigma_i}{\partial \theta_j} \\
= &\sum_{i=1}^L\left[(\sigma_i - y_i)x_{i,j}\right] + \lambda\theta_j - \beta\,\tilde{p}\sum_{i=L+1}^{L+U}\left[(1-\sigma_i)x_{i,j}\right]
\end{aligned}
$$
$$(22)$$

Algorithm 1 CSL Learning Procedure

Input: Feature matrix $\mathbf{X}$, actual labels $\mathbf{Y}_L$, multigraph $\mathbf{G}$
Output: Model parameters $\Theta = \{\theta_j\} \cup \{\alpha_{m,j}\}$
1: Initialize all parameters θ_j and $\alpha_{m,j}$ to zero
2: **for** each graph $m \in \{1, \ldots, M\}$ **do**
3:     Construct the relational features $\pi_{m,i,j}$ using (12)
4: **end for**
5: **repeat**
6:     Generate the prediction scores σ_i based on (11)
7:     Compute the overall loss $\mathcal{L}$ using (20)
8:     Compute the slopes $\frac{\partial \mathcal{L}}{\partial \theta_j}$ (as well as $\frac{\partial \mathcal{L}}{\partial \alpha_{m,j}}$) via (22)
9:     Perform an L-BFGS iteration using $\mathcal{L}$ and $\frac{\partial \mathcal{L}}{\partial \theta_j}$ $\left(\frac{\partial \mathcal{L}}{\partial \alpha_{m,j}} \right)$
10: **until** stopping criterion is met

TABLE 1
Count statistics of our SGTwitter data

Entity	Mean	Median	25%	75%	Maximum
#tweets	168.45	48	12	173	16,888
#followees	349.99	198	109	342	512,978
#followers	852.32	174	80	333	4,062,786
#mentions	123.41	26	4	117	37,457
#retweets	54.12	7	0	39	16,860

$25\% : 25^{th}$ percentile, $75\% : 75^{th}$ percentile

TABLE 2
Label distribution of SGTwitter data

Account type		Marital status	
Personal	794	Single	1,304
Organization	514	Married	1,009
Unlabeled	99,189	Unlabeled	98,184

Finally, we may differentiate (22) to obtain the curvature:

$$\frac{\partial^2 \mathcal{L}}{\partial \theta_j^2} = \sum_{i=1}^{L} \left[\sigma_i \left(1 - \sigma_i\right) x_{i,j}^2 \right] + \lambda + \beta \, \tilde{p} \sum_{i=L+1}^{L+U} \left[\sigma_i \left(1 - \sigma_i\right) x_{i,j}^2 \right] \tag{23}$$

It is clear that the curvature (23) will always be positive for any positive λ (since $\sigma_i \in [0,1]$). We can thus conclude that the full loss $\mathcal{L}$ is *strictly convex* for $\lambda > 0$. The convexity for the relational weight $\alpha_{m,j}$ can be proven in the same manner, and thus we omit the details here for brevity.

Algorithm. Thanks to the convexity trait, we can use any off-the-shelf gradient-based algorithm to learn the parameters of our CSL model. In this work, we use the *limited memory Broyden–Fletcher–Goldfarb–Shanno* (L-BFGS) algorithm [22], a popular, efficient Quasi-Newton method for solving unconstrained optimization problems[2]. Algorithm 1 outlines the overall CSL learning procedure, combining the MRF and CD formulations. As for the stopping criterion, we terminate the algorithm when a maximum iteration I (default: 10) is reached, or the maximum projected slope is below a threshold ϵ (default: 10^{-5}). Notably, the memory/time complexity of the L-BFGS algorithm is linear in the problem size [22], and the convexity of our CSL formulation makes it possible to reach the optimum within a few iterations.

4 USER PROFILING IN TWITTER

This section provides an overview of the Twitter dataset and user profiling tasks we consider in this work.

4.1 Twitter Dataset

In our study, we use the Twitter data of Singapore users—hereafter called *SGTwitter*–collected from the period of 1–28 February 2014. Starting from a set of seed Singapore users, we crawled their network based on the follow, retweet, and user mention links. Next, we added to our user base those followers/followees, retweet sources, and mentioned users who declare Singapore as their profile location. Accordingly, we obtained a total of 130,142 public user accounts whose profiles can be accessed/studied. We then focused on active users who tweeted at least twice within 1 month, which gave us the final set of 100,497 active users.

Table 1 summarizes the count statistics of our SGTwitter data for different activities, aggregated at the user level. In general, we can see that the activity counts follow a heavy-tail distribution. As expected, other than celebrity users, a user generally follows more users than being followed. Intuitively, a user could select followees he/she is interested in, but not the followers. Hence, we can expect the followee links to be a better representation of user interests than the follower links. On the other hand, we can see that user mention and retweet activities are much more focused/targeted, resulting in sparser connectivity in the mention and retweet graphs than in the follow graph. We shall focus on the followee, mention and retweet links in our studies later.

4.2 User Attributes

In this work, we consider the task of classifying two user attributes (i.e., labels): *account type* (i.e., *personal* vs. *organization*) and *marital status* (i.e., *single* vs. *married*)[3]. Profiling these attributes is a relatively new problem that has not been well studied before. This could bring about benefits in terms of providing customized services/supports that cater for the different needs of each user type. For example, organization accounts may require a service to standardize the format of their content postings or to track sentiments on their products, whereas personal accounts would likely benefit from personalized friend and content recommendation. Similarly, married users would likely be more interested in family-related products or contents than single users.

To derive the account type and marital status labels, we first defined several keywords/phrases describing the respective labels. For the account type task, we detected organization accounts by checking if the URLs in their profile description end with "com.sg", "edu.sg" and "gov.sg". We randomly sampled the remaining Twitter users, and then manually labeled and judged if they are personal accounts. For the marital status, examples of relevant keywords/phrases are "wife", "spouse", "my son" for married users, and "girlfriend", "in a relationship", "my gf" for single (unmarried) users. After identifying accounts with the relevant keywords/phrases in their profile description, we manually verified the label assignment of each account.

---

2. More specifically, we use an implementation of the L-BFGS algorithm provided in the SciPy library: http://goo.gl/q2dfnZ

3. While our work currently focuses on two user attributes, we note that CSL is general and readily applicable to any attributes/labels.

Table 2 summarizes the label distribution of our SGTwitter data. Here the minority class labels are "organization" and "married" for the account type and marital status tasks, respectively. Our main interest is to correctly predict these minority cases, which are expected to form a small portion in the complete data, and are thus harder to predict.

4.3 Feature Extraction

Our primary interest here is to investigate to what extent the contents generated by a user can be used to infer his/her (latent) attributes. As such, this work shall be focused primarily on *text features* derived from users' tweets, though we note that our CSL approach is generic and can work on any type of features. We use the term *document* to refer to a data instance i, which represents the collection of tweets posted by a user i. In this context, our goal is to infer a user's attribute based on his/her tweet document.

To extract the text features, we first converted the raw tweets into a *bag-of-words* vector, from which we can derive an n-gram representation suitable for our CSL model. We summarize our feature extraction steps as follows:

- *Tokenization*: We broke a tweet document into its constituent word tokens, and then created bags of word tokens, where each bag has the frequency of the tokens appearing in a document. Prior to tokenization, we also converted all letters to lowercase and devised regular expressions to extract and retain special entities such as emoticons, HTML/URL tags, phone numbers, and hashtags.
- *Stop-word removal*: We then omitted words that appear very frequently and contribute little to discriminating the tweets of a user from those of another user. We used the list of English stop-words in [19].
- *Normalization*: To normalize the word frequencies, we applied the *term frequency–inverse document frequency* (TF-IDF) scheme [3], which puts greater importance on words that appear frequently in a document, and deems words that occur in many documents as less important. Our TF-IDF vectors comprise 1-gram, 2-gram, and 3-gram representations [3].

4.4 Multi-Relational Information

In our study on the SGTwitter data, we consider the multi-relational information **G** derived from three directed graphs: the *follow*, *mention*, and *retweet* graphs. The follow graph contains binary edge weights. That is, $w_{m,i,i'} = 1$ if a user i follows another user i' and 0 otherwise. On the other hand, the weights of the mention/retweet graph refer to the number of times (count) of a user i mentioning/retweeting user i'. In this case, no edge is constructed for a zero count. For each user, we consider his/her *out-edges* in order to compute the relational features $\pi_{m,i,j}$ for all three graphs.

5 EXPERIMENTAL RESULTS AND ANALYSIS

This section presents the results of our study on profiling the account type and marital status of the SGTwitter users. We aim at addressing several research questions (RQs):

- *RQ1*: How does the performance of CSL compare with that of other SSL methods?

- *RQ2*: How do multi-relational features and unlabeled data contribute to the performance?
- *RQ3*: What are the important features and relationship types for predicting user labels?
- *RQ4*: How well can the learned CSL model generalize to unseen (unlabeled) data?
- *RQ5*: What can predictions made by CSL tell about a larger user population?

Procedure. To address the above RQs, we consider two scenarios: evaluation using labeled data (for *RQ1–RQ3*), and evaluation using unlabeled data (for *RQ4–RQ5*). For the first scenario, we adopt a *stratified* 10-fold CV procedure, i.e., we split the SGTwitter data into 10 sets of training and testing data, each retaining the class label proportion as per the original data. We then report the averaged performance. Also, for all SSL methods considered in this study, their training for each fold involves using labeled instances in the training set, plus *all* the remaining unlabeled instances from the full original data. For the second scenario, we manually inspect the top k-predicted users for each class label. The goal is to see how well our method predicts on completely unseen data (i.e., not labeled apriori), and compare the predictions with the labeled dataset.

Metric. To quantify performance, we examine the averaged *F1-score*, an evaluation metric that is popularly used in text classification and information retrieval [3]. The *F1-score* measures classification accuracy in terms of a harmonic mean of $Precision$ and $Recall$, i.e., $F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$, where $Precision = \frac{TP}{TP+FP}$, $Recall = \frac{TP}{TP+FN}$, and TP, FP and FN are the true positives, false positives, and false negatives respectively. For these metrics, again the positive class refers to the minority labels, i.e., "organization" for account type and "married" for marital status. Lastly, we also look at the averaged *training time* (in seconds) of a given method, so as to gauge its computational efficiency.

Baseline. We compare our CSL method with several representative SSL baseline methods. The first baseline is *bootstrapping* [12], where we first train a logistic regression using labeled data, apply it to predict on unlabeled data, and then add into the labeled dataset those samples that have high prediction scores. We repeat this for 10 iterations, where for each iteration $n \in \{1, \dots, 10\}$, we add the top $\frac{n}{10}$ predicted samples into the labeled set. The second baseline is *label spreading* (LS), a popular graph-based SSL method [36]. We reiterate from Section 2.2 that LS relies on the affinity assumption, and its success depends on the choice of affinity matrix. We explore two renowned variants of kernel functions to define the affinity matrix in LS: k-nearest-neighbor (kNN) kernel, and radial basis (RBF) kernel [36].

Additionally, we compare our CSL approach with two state-of-the-art information theoretic SSL methods, namely *entropy regularization* (ER) [11] and *expectation regularization* (XR) [23]. The overall loss functions $\mathcal{L}$ for the ER and XR methods are respectively as follows:

$$\text{ER} : \mathcal{L} = - \sum_{i=1}^{L} [y_i \ln(\sigma_i) + (1 - y_i) \ln(1 - \sigma_i)]$$
$$- \beta \sum_{i=L+1}^{L+U} \sigma_i \ln(\sigma_i) \qquad (24)$$

$$\text{XR}: \mathcal{L} = -\sum_{i=1}^{L}\left[y_i \ln(\sigma_i) + (1-y_i)\ln(1-\sigma_i)\right]$$
$$+ \frac{\lambda}{2}\sum_{j=1}^{J}\left(\theta_j^2 + \sum_{m=1}^{M}\alpha_{m,j}^2\right) - \beta \ln\left(\tilde{p}\sum_{i=L+1}^{L+U}\sigma_i^2\right) \quad (25)$$

Configuration. For the ER, XR, and CSL methods, we chose the best SSL regularization parameter β from the following candidate list: $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 100\}$. Meanwhile, the L2 regularization parameter for XR and CSL was fixed to $\lambda = 1$, which we found to give good results overall. We also note that all experiments presented in this paper were carried out on a computer server with the following virtual machine configuration: 7-core Intel Xeon 2.6 GHz processor with a total of 70 GB memory (RAM).

Significance test. To evaluate whether the performance difference between two methods is statistically significant, we perform the *Wilcoxon signed-rank test* [34] with a critical value of 0.05. The Wilcoxon test provides a non-parametric alternative to the t-test for matched pairs, when the pairs cannot be assumed to be normally distributed. When the test yields a p-value less than 0.05, we deem that the performance difference is significant (and vice versa).

5.1 Performance Comparisons (RQ1)

We first evaluate the 10-fold CV results of our method using the "full" training set. Specifically, for each fold, we train our CSL method using 90% of the available labeled data (i.e., 90% of 1,308 and 2,313 labeled accounts for account type and marital status tasks respectively; cf. Table 2), plus all the remaining unlabeled data. Tables 3 and 4 show the results for the two tasks respectively, comparing the *F1-scores* and training time of our CSL method with those of all the other SSL methods. To facilitate comprehensive evaluations, we present the results for different MRF settings: no graph, single graph, and all three graphs. In addition, we show the p-values of the Wilcoxon test for comparing the *F1-scores* of the respective MRF settings (e.g., the p-value for bootstrapping with *follow graph* features involves comparison to CSL with *follow graph* features as well). As an additional reference, we include the *F1-score* produced by a random guess[4]

From Tables 3 and 4, we can see that our CSL method significantly outperforms the bootstrapping and LS methods, in terms of both *F1-score* and training time. This is true for all graph configurations. The *F1-score* of our method is also substantially better than that of the random guess baseline. It can be seen here that the "high-confidence prediction is correct" assumption of the bootstrapping method leads to a suboptimal performance (especially for the marital status task). It is also evident that the affinity assumption of the LS methods is inappropriate for our profiling tasks. In sum, these show that incorrect assumption about the data/task at hand in SSL can lead to mistakes that reinforce themselves.

Additionally, the results show that in general our CSL approach compares favourably to the ER and XR methods in terms of *F1-score*, although there are cases where the performance difference is marginal (i.e., p-value ≥ 0.05). Regardless, we will show later in a further sensitivity study

4. Random guess refers to the case where $\frac{TP}{TP+FN} = \frac{FP}{FP+TN}$.

TABLE 3
Classification results for account type classification

Method	Graph	F1-Score	Time (sec)	p-value
Random guess baseline	None	0.3930	—	0.0025∗
Bootstrapping	None	0.8275	29.4142	0.0035∗
	Follow	0.8441	76.1515	0.0297∗
	Mention	0.8273	56.0481	0.0109∗
	Retweet	0.8264	70.8066	0.0142∗
	All	0.8373	113.2009	0.0047∗
Label spreading (kNN)	None	0.5311	30.7675	0.0025∗
	Follow	0.5588	60.4088	0.0025∗
	Mention	0.5548	57.5618	0.0025∗
	Retweet	0.5282	58.8224	0.0025∗
	All	0.5477	114.2311	0.0025∗
Label spreading (RBF)	None	0.6572	5.4188	0.0025∗
	Follow	0.7503	6.6155	0.0025∗
	Mention	0.6496	6.4276	0.0025∗
	Retweet	0.6504	6.4941	0.0025∗
	All	0.7318	8.9634	0.0025∗
Entropy regularization	None	0.8230	1.0824	0.0372∗
	Follow	0.8481	4.2607	0.0372∗
	Mention	0.8169	3.1640	0.0083∗
	Retweet	0.8294	2.2800	0.0463∗
	All	0.8494	7.8640	0.1013
Expectation regularization	None	0.8490	0.8710	0.1587
	Follow	0.8643	3.6370	0.1587
	Mention	0.8454	3.1392	0.0398∗
	Retweet	0.8438	2.6258	0.1587
	All	0.8563	7.9321	0.0544
Our approach (CSL)	None	0.8481	0.9972	—
	Follow	0.8635	4.4847	—
	Mention	0.8491	3.1003	—
	Retweet	0.8453	2.6968	—
	All	0.8608	8.8409	—

∗: significant at a critical value of 0.05; −: not applicable

TABLE 4
Classification results for marital status classification

Method	Graph	F1-Score	Time (sec)	p-value
Random guess baseline	None	0.4362	—	0.0025∗
Bootstrapping	None	0.5420	20.4187	0.0025∗
	Follow	0.5824	171.1721	0.0025∗
	Mention	0.5842	55.4378	0.0025∗
	Retweet	0.5554	48.2193	0.0025∗
	All	0.6278	356.1654	0.0035∗
Label spreading (kNN)	None	0.5673	35.5445	0.0035∗
	Follow	0.4575	66.9595	0.0025∗
	Mention	0.5625	65.0869	0.0035∗
	Retweet	0.5740	65.8956	0.0035∗
	All	0.4727	133.1662	0.0025∗
Label spreading (RBF)	None	0.5215	5.3937	0.0025∗
	Follow	0.1822	6.6723	0.0025∗
	Mention	0.5509	6.8228	0.0035∗
	Retweet	0.5604	6.5979	0.0025∗
	All	0.2550	9.2841	0.0025∗
Entropy regularization	None	0.5757	1.0106	0.0025∗
	Follow	0.6200	4.4185	0.0025∗
	Mention	0.5906	3.4064	0.0025∗
	Retweet	0.5833	2.3030	0.0025∗
	All	0.6225	8.1122	0.0025∗
Expectation regularization	None	0.5821	1.0399	0.0025∗
	Follow	0.6533	4.1410	0.1206
	Mention	0.6032	2.7075	0.0083∗
	Retweet	0.5966	2.4442	0.0035∗
	All	0.6513	7.7271	0.0463∗
Our approach (CSL)	None	0.6310	1.0050	—
	Follow	0.6644	4.4516	—
	Mention	0.6265	3.1527	—
	Retweet	0.6318	2.4739	—
	All	0.6631	9.6041	—

∗: significant at a critical value of 0.05; −: not applicable

(to be presented in Section 5.2) that CSL is significantly more robust than the two methods. As for the training time, Tables 3 and 4 show that CSL is as efficient as the ER and XR methods, but is an order of magnitude faster than the bootstrapping and LS methods. Finally, comparing the different graph configurations, we find that incorporating
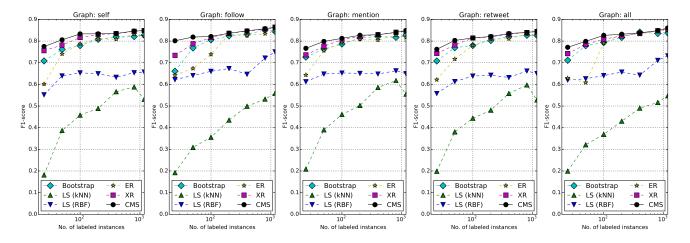
Fig. 2. Classification results for account type task with varying labeled data sizes and graph configurations
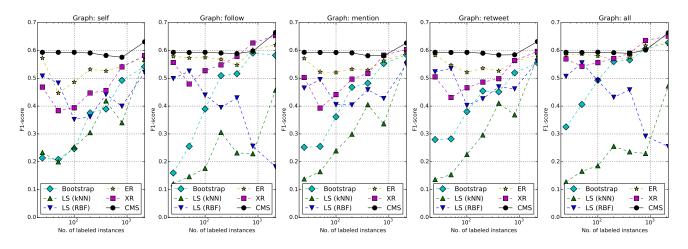

Fig. 3. Classification results for marital status task with varying labeled data sizes and graph configurations

relational features from the follow graph alone already leads to *F1-scores* comparable to those using all the three graphs. This implies that the retweet and mention features are not as useful as the follow features. Nevertheless, unlike the LS methods (especially for the marital status task), using all three graphs in CSL does not significantly degrade the *F1-scores* (compared to using the follow graph alone). This suggests that CSL can make better use of MRF, even when noisy or less relevant relational features are used.

5.2 Contribution of MRF and Unlabeled Data (RQ2)

To see the contributions of the MRF and CD formulations, further sensitivity studies were carried out by varying the graph configurations and number of labeled instances in the training data, respectively. We chose the number of labeled instances L from $\{25, 50, 100, 200, 400, 800, \text{"full"}\}$, where "full" refers to 90% of all the labeled instances, as already explained in Section 5.1. Fig. 2 and 3 consolidate the results of our studies for account type and marital status tasks respectively. Note that the results at the right-hand extremes of the figures correspond those in Tables 3 and 4.

We can see here that the CD regularization in CSL yields more robust and consistent *F1-scores* than the ER and XR

regularization, especially for small labeled data size. For example, even when $L = 25$, CSL is able to achieve *F1-score* not far from that obtained with $L = \text{"full"}$, for both profiling tasks. This is also verified by our Wilcoxon tests, all of which yielded p-values less than 0.05. Moreover, our CSL method performs more consistently than the other methods under different graph configurations. As before, the retweet and mention features contribute less compared to the follow features. Finally, Fig. 4 and 5 present the breakdown of training times of different methods. It is again shown that CSL is on par with ER and XR, but substantially faster than the bootstrapping and LS methods. All in all, these show that CSL exhibits both robustness and efficiency, making it more preferable than other methods for profiling tasks.

5.3 Feature Importance Analysis (RQ3)

We can now probe into the parameters of the trained CSL model, and investigate in details which features are the most discriminative for our user profiling tasks. Specifically, we assess the importance of the individual features by looking at the learned self weights θ_j and relational weights $\alpha_{m,j}$. Fig. 6 and 7 present the top 15 features (i.e., having the largest *absolute values* of θ_j and $\alpha_{m,j}$) for the account type
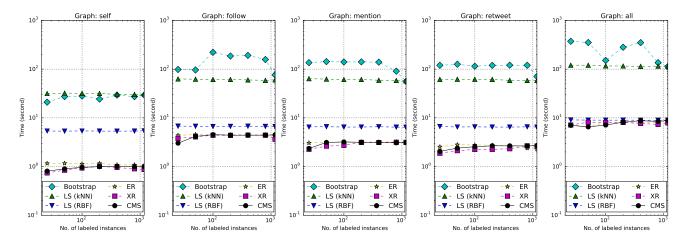
Fig. 4. Training time for account type with varying labeled data sizes and graph configurations
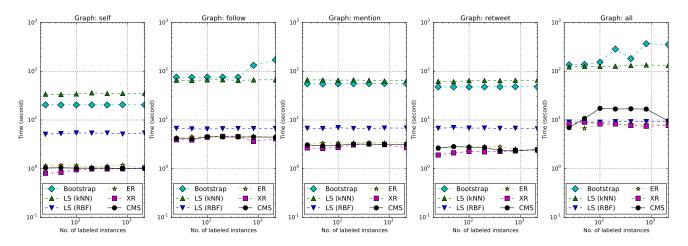

Fig. 5. Training time for marital status with varying labeled data sizes and graph configurations

and marital status profiling tasks respectively. The leftmost bar chart in each figure shows the self weights θ_j, while the remaining charts show the relational weights $\alpha_{m,j}$ for the follow, mention, and retweet graphs respectively. We also note that a positive weight suggests that the respective feature is correlated with the positive label (i.e., "organization" or "married"), whereas a negative weight corresponds to the negative label (i.e., "personal" or "single").

The results reveal several interesting insights that conform with our intuition. For example, in the account type profiling task, we can see that informal expressions such as "just", "lol" or "haha" are correlated with personal accounts, whereas organizations tend to be associated with more neutral words such as "singapore", "new", or "cny" (acronym for "Chinese New Year"). Similarly for the marital status task, family-related words such as "god", "great" or "kids" are often correlated with married people, whereas single people consist of young adults and students, who like to use words such as "school", "shit", etc. In general, we also find that the feature correlation is consistent across different feature groups (e.g., when "singapore" in self features correlates with the organization account, the correlation also holds for "singapore" in mention features).

Furthermore, it is evident from the overall weight mag-nitudes in Fig. 6 and 7 that the self features are the most discriminative, though the other, relational features are still collectively useful. Among the three groups of relational features, we find that those derived from the follow graph are the most relevant. This can be attributed to the fact that the follow graph is more dense than the retweet or mention graph (see Table 1), thus yielding more informative features.

To examine the validity of the learned top features, we grouped all (labeled) instances according to the labels and examined the feature distributions for each label. Fig. 8 shows the cumulative distribution function (CDF) of the top positive and negative features for each feature group in the account type task. An early increase in the CDF value implies a more skewed feature distribution, according to which we can then judge whether the learned positive or negative correlation is valid. For instance, the CDF of the word "singapore" is less skewed for organization accounts, suggesting that organizations use the word more often than personal accounts. Comparing Fig. 6 with Fig. 8, we can conclude that, overall, the feature distributions conform with the learned feature correlations. The conclusion also holds for all the remaining features as well as the features for the marital status task. Owing to space constraints, however, we are not able to show the exhaustive results here.
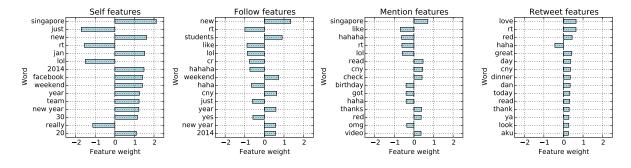
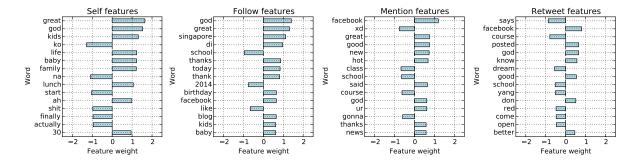Fig. 6. Feature importance for account type classification

Fig. 7. Feature importance for marital status classification

5.4 Generalization to Unseen Data (RQ4)

Further studies have been conducted to evaluate the ability of our learned model to generalize to novel, unlabeled data. To this end, we used our trained CSL model to predict for all unlabeled data, and picked the top K positive instances with the highest prediction scores σ_i, as well as the top K negative instances with the highest $(1 - \sigma_i)$. We then manually inspected all these instances to see how well the CSL predictions match with human judgments. To assess the model robustness, we varied K from 20 to 100. For each class label, we recorded the number of correctly predicted instances (TP), that of incorrectly predicted instances (FP), and that of unclassifiable instances (UC). The UC refers to the case whereby we are unable to manually determine the actual label of an instance (i.e., a "don't know" answer). We then computed the precision at top K ($Prec$) by excluding the unclassifiable instances, i.e., $Prec = \frac{TP}{K - UC}$.

Tables 5 and 6 summarize the results for account type and marital status tasks respectively. Overall, we can see that the learned CSL model can predict for both the account type and marital status pretty well, which is evident from the fairly low FP and high $Prec$ results. Meanwhile, the UC numbers are generally low, except for the "married" label of the marital status task. The latter suggests that determining whether a person is married is a difficult task, even for humans. Nonetheless, the overall results demonstrate that our method has good generalization abilities.

5.5 Inference on Larger User Population (RQ5)

The final part of our empirical studies consists of making inference on a larger population of SGTwitter users. To this end, we carried out quantitative and qualitative analyses on
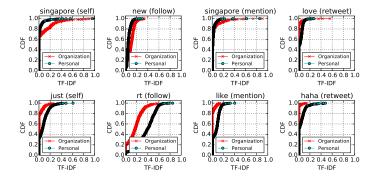

Fig. 8. Cumulative distribution of the top features for account type task

the predictions made for all unlabeled data. Our quantitative study involves comparing the label distributions in the labeled dataset (as per Table 2) with the predicted distributions as inferred by CSL on the unlabeled dataset. Table 7 shows the results. We find that the predicted distributions are much more imbalanced than the distributions of the labeled data. Although we cannot fully verify this observation (due to the need to label all 100K samples), it is reasonable to expect that the larger SGTwitter population would consist of more personal accounts than organization ones, and more single users than married ones. The distribution difference also suggests that the larger population contains new cases that are not previously captured in the labeled data, and our method can generalize to these cases fairly well.

Further qualitative analysis on the individual users reveals additional insights about the (larger) SGTwitter population. Fig. 9 shows the screenshot of a top-predicted organization account that is not previously captured in the

TABLE 5
Top K predictions on unseen data instances for account type task

Label	Metric	Top 20	Top 40	Top 60	Top 80	Top 100
Organization	TP	18	36	56	72	89
	FP	2	4	4	8	11
	UC	0	0	0	0	0
	$Prec$	0.9000	0.9000	0.9333	0.9000	0.8900
Personal	TP	18	37	55	73	90
	FP	0	0	0	0	0
	UC	2	3	5	7	10
	$Prec$	1.0000	1.0000	1.0000	1.0000	1.0000

TP: true positives, FP: false positives, UC: unclassifiable, $Prec$: precision

TABLE 6
Top K predictions on unseen data instances for marital status task

Label	Metric	Top 20	Top 40	Top 60	Top 80	Top 100
Married	TP	9	19	24	33	39
	FP	0	0	2	4	6
	UC	11	21	33	42	54
	Prec	1.0000	1.0000	0.9231	0.8919	0.8667
Single	TP	16	34	53	69	88
	FP	0	0	0	0	0
	UC	4	6	7	11	12
	Prec	1.0000	1.0000	1.0000	1.0000	1.0000

TP: true positives, FP: false positives, UC: unclassifiable, $Prec$: precision

TABLE 7
Comparison of label distributions

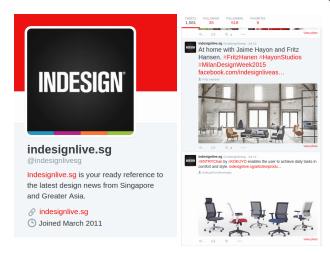| Dataset	Account type		Marital status	
	Label	#users	Label	#users
Labeled	Personal	794 (60.7%)	Single	1,304 (56.4%)
	Organization	514 (39.3%)	Married	1,009 (43.6%)
Unlabeled	Personal	88,101 (88.0%)	Single	62,596 (62.4%)
	Organization	12,396 (12.0%)	Maried	37,901 (37.6%)

Fig. 9. Example of top organization account

networks (e.g., Facebook, Foursquare, etc.). We also plan to develop a multi-task learning framework that can infer multiple user attributes jointly by modeling their dependencies (e.g., correlation between age group and marital status).

ACKNOWLEDGMENTS

labeled dataset. In particular, the profile description of the account has an URL with a new suffix ".sg", which is not part of the suffices used to derive the labeled dataset (i.e., ".com.sg", ".edu.sg" and ".gov.sg"; see Section 4.2). This shows that our CSL method is able to properly predict for novel instances, based on content (word) features alone. Similarly for the marital status task, we found several interesting insights (not shown here due to space limitation). For instance, some of the top-predicted married users never use the keywords/phrases listed for the labeled data (see again Section 4.2) in their profile descriptions, but their profile pictures clearly show that they have a spouse or children.

6 CONCLUSION

In this paper, we put forward a novel CSL approach for modeling/profiling the attributes of users in social media. The proposed approach provides a principled and efficient solution to the novel problem of simultaneously exploiting multiple types of social relationship and large pool of unlabeled data in user profiling tasks. The centerpiece of the proposed CSL approach is to first expand the input space by generically constructing a set of MRF features that capture different types of relationship, and then perform the CD regularization to establish convex semi-supervised learning using unlabeled data. The experimental results on Singapore Twitter users have demonstrated the accuracy, robustness, efficiency, as well as interpretability traits of our approach in profiling the user attributes.

Moving forward, we wish to extend our methodology to more challenging profiling tasks involving multiple social

REFERENCES

[1] S. Abney, "Understanding the Yarowsky algorithm," *Computational Linguistics*, vol. 30, no. 3, pp. 365–395, 2004.

[2] S. M. Ali and S. D. Silvey, "A General Class of Coefficients of Divergence of One Distribution from Another," *Journal of the Royal Statistical Society*, vol. 28, no. 1, pp. 131–142, 1966.

[3] R. A. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*. Addison-Wesley, 1999.

[4] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of Machine Learning Research*, vol. 7, pp. 2399–2434, 2006.

[5] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of the Annual Conference on Computational Learning Theory*, 1998, pp. 92–100.

[6] O. Chapelle, B. Schlkopf, and A. Zien, *Semi-Supervised Learning*. The MIT Press, 2006.

[7] W. Chu and Z. Ghahramani, "Probabilistic models for incomplete multi-dimensional arrays," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2009, pp. 89–96.

[8] Y. Dong, Y. Yang, J. Tang, Y. Yang, and N. V. Chawla, "Inferring user demographics and social strategies in mobile social networks," in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2014, pp. 15–24.

[9] L. R. Drumond, E. Diaz-Aviles, L. Schmidt-Thieme, and W. Nejdl, "Optimizing multi-relational factorization models for multiple target relations," in *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2014, pp. 191–200.

[10] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[11] Y. Grandvalet and Y. Bengio, "Entropy regularization," in *Semi-Supervised Learning*, O. Chapelle, B. Schölkopf, and A. Zien, Eds., 2006, pp. 151–168.

[12] G. Haffari and A. Sarkar, "Analysis of semi-supervised learning with the Yarowsky algorithm," in *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2007, pp. 159–166.

[13] R. A. Harshman and M. E. Lundy, "PARAFAC: Parallel factor analysis," *Computational Statistics and Data Analysis*, vol. 18, no. 1, pp. 39–72, 1994.

[14] K. Ikeda, G. Hattori, C. Ono, H. Asoh, and T. Higashino, "Twitter user profiling based on text and community mining for market analysis," *Knowledge-Based Systems*, vol. 51, pp. 35–47, 2013.

[15] R. Jenatton, N. L. Roux, A. Bordes, and G. Obozinski, "A latent factor model for highly multi-relational data," in *Advances in Neural Information Processing Systems*, 2012, pp. 3176–3184.

[16] J. L. W. V. Jensen, "Sur les fonctions convexes et les ingalits entre les valeurs moyennes," *Acta Mathematica*, vol. 30, no. 1, pp. 175–193, 1906.

[17] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proceedings of the International Conference on Machine Learning*, 1999, pp. 200–209.

[18] M. Kosinski, D. Stillwell, and T. Graepel, "Private traits and attributes are predictable from digital records of human behavior," *Proceedings of the National Academy of Sciences*, vol. 110, no. 15, pp. 5802–5805, 2013.

[19] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li, "RCV1: A new benchmark collection for text categorization research," *Journal of Machine Learning Research*, vol. 5, pp. 361–397, 2004.

[20] J. Li, A. Ritter, and E. Hovy, "Weakly supervised user profile extraction from Twitter," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2014.

[21] R. Li, C. Wang, and K. C.-C. Chang, "User profiling in an ego network: Co-profiling attributes and relationships," in *Proceedings of the International World Wide Web Conference*, 2014, pp. 819–830.

[22] D. C. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," *Mathematical Programming*, vol. 45, no. 3, pp. 503–528, 1989.

[23] G. S. Mann and A. McCallum, "Simple, robust, scalable semi-supervised learning via expectation regularization," in *Proceedings of the International Conference on Machine Learning*, 2007, pp. 593–600.

[24] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, "You are who you know: Inferring user profiles in online social networks," in *Proceedings of the ACM International Conference on Web Search and Data Mining*, 2010, pp. 251–260.

[25] T. Mitchell, "The role of unlabeled data in supervised learning," in *Language, Knowledge, and Representation*, J. Larrazabal and L. Miranda, Eds. Springer, 2004, vol. 99, pp. 103–111.

[26] M. Nickel, V. Tresp, and H.-P. Kriegel, "Factorizing YAGO: scalable machine learning for linked data," in *Proceedings of the International World Wide Web Conference*, 2012, pp. 271–280.

[27] G. Niu, W. Jitkrittum, B. Dai, H. Hachiya, and M. Sugiyama, "Squared-loss mutual information regularization: A novel information-theoretic approach to semi-supervised learning," in *Proceedings of the International Conference on Machine Learning*, 2013, pp. 10–18.

[28] D. Rao, D. Yarowsky, A. Shreevats, and M. Gupta, "Classifying latent user attributes in Twitter," in *Proceedings of the International Workshop on Search and Mining User-Generated Contents*, 2010.

[29] A. P. Singh and G. J. Gordon, "Relational learning via collective matrix factorization," in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2008, pp. 650–658.

[30] C. Sutton and A. McCallum, "An introduction to conditional random fields," *Foundations and Trends in Machine Learning*, vol. 4, no. 4, pp. 267–373, 2012.

[31] T. Suzuki, M. Sugiyama, T. Kanamori, and J. Sese, "Mutual information estimation reveals global associations between stimuli and biological processes," *BMC Bioinformatics*, vol. 10, 2009.

[32] J. Wang, H. Bensmail, and X. Gao, "Multiple graph regularized protein domain ranking," *BMC Bioinformatics*, vol. 13, no. 1, p. 307, 2012.

[33] M. Wang, X.-S. Hua, R. Hong, J. Tang, G.-J. Qi, and Y. Song, "Unified video annotation via multigraph learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 5, pp. 733–746, 2009.

[34] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.

[35] Z. Xu, K. Kersting, and V. Tresp, "Multi-relational learning with gaussian processes," in *Proceedings of the International Jont Conference on Artifical Intelligence*, 2009, pp. 1309–1314.

[36] D. Zhou, B. Olivier, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Advances in Neural Information Processing Systems*, 2004, pp. 321–328.

Richard J. Oentaryo is a Research Scientist at the Living Analytics Research Centre, School of Information Systems, Singapore Management University. He obtained his Ph.D. and B.Eng. (First Class Honour) in Computer Engineering from the Nanyang Technological University, Singapore, in 2011 and 2004 respectively. His research interests include machine learning, data mining, and nature-inspired computing. Dr. Oentaryo has published more than 25 papers in various international journals and conferences.

Ee-Peng Lim is a Full Professor at the School of Information Systems, Singapore Management University. He received Ph.D. from the University of Minnesota, Minneapolis in 1994. His research interests include social network and web mining, information integration, and digital libraries. He has published more than 300 papers at international journals and conferences. He currently serves as an Associate Editor of the IEEE Transactions on Knowledge and Data Engineering (TKDE), ACM Transactions on Information Systems (TOIS), ACM Transactions on the Web (TWeb), Information Processing and Management (IPM), Social Network Analysis and Mining (SNAM), and Journal of Web Engineering (JWE).

Freddy Chong Tat Chua is a Research Scientist at the Mechanisms and Design Lab of Hewlett Packard Enterprise Labs based in Palo Alto, California, USA. He is also an Adjunct Faculty in the School of Information Systems, Singapore Management University (SMU). Dr. Chua obtained Ph.D. in Information Systems from SMU in 2013, and Bachelor of Computer Science from National University of Singapore in 2007. He serves as a Program Committee member at several academic communities. He has published extensively in prominent computer science conferences and journals.

Jia-Wei Low is a Data Scientist at the Infocomm Development Authority (IDA), Singapore. Prior to IDA, he worked as a Research Engineer at the Living Analytics Research Centre, School of Information Systems, Singapore Management University. He co-founded a technology startup company specializing in electricity monitoring and home automation in 2011. His recent works include predictive modelling for public healthcare sector and emergency response.

David Lo is an Assistant Professor at the School of Information Systems, Singapore Management University. He works in the area of data mining and software engineering. His primary research interests include frequent pattern mining, social network mining, dynamic program analysis, and specification mining. He received a Ph.D. in computer science from the National University of Singapore in 2008.