# A Community-Based Pseudolikelihood Approach for Relationship Labeling in Social Networks

Huaiyu Wan*, Youfang Lin*, Zhihao Wu, and Houkuan Huang

School of Computer and Information Technology,
Beijing Jiaotong University, Beijing 100044, China
{huaiyuwan,yflin,zhihaowu,hkhuang}@bjtu.edu.cn

**Abstract.** A social network consists of people (or other social entities) connected by a set of social relationships. Awareness of the relationship types is very helpful for us to understand the structure and the characteristics of the social network. Traditional classifiers are not accurate enough for relationship labeling since they assume that all the labels are independent and identically distributed. A relational probabilistic model, relational Markov networks (RMNs), is introduced to labeling relationships, but the inefficient parameter estimation makes it difficult to deploy in large-scale social networks. In this paper, we propose a community-based pseudolikelihood (CBPL) approach for relationship labeling. The community structure of a social network is used to assist in constructing the conditional random field, and this makes our approach reasonable and accurate. In addition, the computational simplicity of pseudolikelihood effectively resolves the time complexity problem which RMNs are suffering. We apply our approach on two real-world social networks, one is a terrorist relation network and the other is a phone call network we collected from encrypted call detail records. In our experiments, for avoiding losing links while splitting a closely connected social network into separate training and test subsets, we split the datasets according to the links rather than the individuals. The experimental results show that our approach performs well in terms of accuracy and efficiency.

**Keywords:** Social networks, Relationship labeling, Community structure, Pseudolikelihood, Conditional random fields.

## 1 Introduction

Social networks are a ubiquitous paradigm of human interactions in real world. People in social networks are connected to each other by different types of relationships, such as family, friendship, co-working, collaboration, contact, etc. Given a snapshot of a social network with content and link structure, can we infer the types of the relationships between the individuals? This question can be formalized as the *relationship labeling problem*. Labeling relationships is one of the most significant problems in the research of social networks. For instance,

* Corresponding authors.

D. Gunopulos et al. (Eds.): ECML PKDD 2011, Part III, LNAI 6913, pp. 491–505, 2011.

in a criminal network, the labels of the relationships between the criminals can help the police to discover regular patterns about the organization and operation of the criminal group. Considering another example of a social network which consists of all the mobile users in a particular region, knowing the type of the relationship between each pair of communicated users can greatly help the mobile service providers to develop targeted marketing strategies.

In many real world applications, a common situation of the relationship labeling task is that, some small part of the relationships in a social network can be directly labeled in some way, while the others cannot be labeled directly and inference is needed. This is so-called *within-network learning*. For example, in a criminal network, some relationships between the criminals can be labeled through the police investigation, while the others must be labeled in other ways. Similarly, in a mobile phone call network, a few relationships between the communicated users can be labeled through the service packages (i.e., family packages or group packages) ordered by the users, but more other relationships cannot be labeled in this way.

The basic idea for classifying the relationships is employing traditional classifiers in the flat setting by using the content attributes of the relationships, where all the labels are assumed to be independent and identically distributed (IID). However, this completely ignores the rich information of the link structure, which generally reflects the common patterns of interactions among the individuals in a social network. Therefore, Taskar et al. [1] and Zhao et al. [2] adopt relational Markov networks (RMNs) [3], a statistical relational learning framework, to classify the relationship labels in webpage networks and terrorist networks respectively, but the inefficient parameter estimation makes it very difficult to deploy this model in large-scale social networks. In addition, the definition of the *relational clique templates* will greatly affect the prediction accuracy of RMNs in practice.

In this paper, we propose a community-based pseudolikelihood (CBPL) approach to labeling relationships in social networks. In our approach, we use the community structure of a social network to assist in constructing the conditional random field (CRF). As we know, community structure is one of the most important properties of complex networks [4]. According to the notion of "birds of a feather flock together", individuals in the same community tend to have the same type, and thus relationships starting from the same individual and terminating in the same community tend to have the same label. Fig. 1 depicts an example fragment of a terrorist social network [2] from the Profiles in Terror (PIT) knowledge base. A dashed ellipse indicates a community, and the various relationship types are distinguished by different line styles and colors. This figure clearly illustrates the correlation between the relationship labels and the community structure of the network.

As an efficient alternative of likelihood, the pseudolikelihood measure [5] is often employed to approximate the joint probability distribution of a collection of random variables with a set of conditional probability distributions (CPDs). This technique effectively resolves the time complexity problem which the RMN

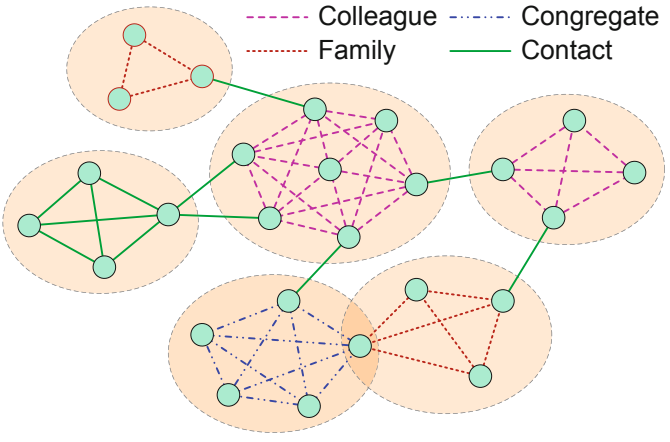


**Fig. 1.** An example fragment of a terrorist social network

model is suffering and makes our approach more efficient for handling large-scale social networks.

We present experiments using our approach on two real-world social networks, one is a terrorist social network [2] and the other is a phone call network we collected from encrypted call detail records (CDRs). A problem we often encounter in the experiments on within-networks is that splitting a closely connected network into separate training and test subsets will lose the information of the links that go from one subset to another. In our experiments, for avoiding losing such information, we split the datasets according to the links (i.e., relationships) rather than the individuals. This ensures that each link will appear in either a training subset or a test subset. The experimental results show that our proposed approach is much more accurate and efficient than the RMN approach on the task of relationship labeling in social networks.

The rest of the paper is organized as follows. The next section provides a brief discussion of related work. Section 3 presents our approach in detail, followed by the experimental evaluations in section 4. Finally, we give the conclusion and future work in section 5.

## 2 Related Work

As discussed earlier, it is not accurate enough for relationship labeling in social networks by only using the content attributes, since the rich information of link structures is completely ignored. Taskar et al. [1] treat the relationship labeling problem as a task of *link prediction* and use RMNs to predict the labels of the links between the Computer Science department webpages from three universities in American. RMNs [3,6] are a joint probabilistic modeling framework for an entire collection of related entities building on undirected graph models, and

provide a form of collective classification in which we can simultaneously decide on the class labels of all the relationships together rather than classify each relationship separately. Zhao et al. [2] pay their attention to the counter-terrorism domain. They extract a terrorist social network from the PIT knowledge base (http://profilesinterror.mindswap.org/) and try to predict the types of the relationships between the terrorists. RMNs are also employed for their experiments. Since two terrorists can be related in multiple relationships, multi-label classification is considered.

There are two problems with using RMNs for labeling relationships in social networks:

- The computational complexity of learning RMNs is very high. That is because multiple rounds of approximate inference (e.g., loopy belief propagation) are required over the entire dataset. Especially in our relationship labeling task, the number of relationships is the squared magnitude of the number of individuals in a social network. So the training time is usually unacceptable if the scale of the social network is too large. In addition, numerous short, closed loops in large-scale RMNs usually cause the belief propagation algorithm to return a poor approximation and even not to converge to a stationary state.
- The prediction accuracy of the RMN model directly depends on the definition of relational clique templates. For relationship labeling, the most direct method is to construct dyad cliques for any pair of relationships which have a common individual and triad cliques for any triple of relationships which connected end to end. However, this will not always be correct in case the individuals in the same clique are not of the same type. So the labeling accuracy will be affected to some extent.

In this work, we propose to use the pseudolikelihood technique to estimate the labels of relationships in social networks. Since pseudolikelihood can only capture the local dependencies and ignores the indirect effects between the non-neighboring variables, it may lose some accuracy in practice. However, we must consider a tradeoff between the prediction accuracy and the computational complexity in relational learning, especially in case the scale of a social network is very large. Actually, as an efficient alternative measure of likelihood, pseudolikelihood has been successfully used in the relational learning field. Richardson and Domingos [7] proposed optimizing a pseudolikelihood measure to learning an Markov logic network (MLN) [8], where the full joint distribution is approximated as a product of each variable's probability conditioned on its Markov blanket. Relational dependency networks (RDNs) [9], an undirected graphical model for relational data introduced by Neville and Jensen, approximate the full joint distribution of an entire dataset with a set of CPDs based pseudolikelihood techniques. Xiang and Neville [10] developed a semi-supervised pseudolikelihood expectation maximization (PL-EM) algorithm, which has been demonstrated to be effective in within-network learning.

Community structure is used in our proposed approach to assist in constructing the CRF of a social network, and we believe that this will amend the limi-

tation of the pseudolikelihood measure and make our approach more reasonable and accurate. The property of community structure has been successfully used to describe the dependencies between the variables in relational data. Neville and Jensen [11] proposed latent group model (LGM), which posits that the class labels of the objects in a relational dataset are related to their group (or community) types. Within each group, the class labels are conditionally independent given the group type. Another relational model similar to LGM is the latent social dimension (LSD) model [12], which extracts latent social dimensions of objects from a modularity matrix defined on the modularity measure [13] and then considers these dimensions as normal features of objects for prediction tasks. The above two models demonstrate that the community structure is really very helpful for relational learning.

Wang et al. [14] studied the mining of advisor-advisee relationship from research publication networks and proposed a time-constrained probabilistic factor graph (TPFG) model, which is an unsupervised approach and suitable for the situation that not any labeled relationships can be used as supervised information.

## 3 Approach

We use a graph $G = (V, E)$ to represent a social network, where $V$ is the set of individuals, and $E$ is the set of links (i.e., relationships) between the individuals. Suppose that the content attributes of the individuals and the link structure are known, and some relationship labels are observed, then our task is to predict the remaining unobserved labels.

In relationship labeling, we need to make relationships the first-class citizens. Given an instantiation $\mathcal{I}$ of our schema, the pseudolikelihood $PL(\mathcal{I})$ is the product of the conditional probability of each variable $Y_i (i \leq |E|)$ given its Markov blanket $MB(Y_i)$. So we need to specify the neighboring relationships for each relationship $e \in E$, i.e., to construct a Conditional random field (CRF) for all the relationships over the entire social network. CRFs are undirected graphical models that were developed for labeling sequence data [15], and are well suited for discriminative training, which generally provides significant improvements in classification accuracy over generative training given sufficient training examples [16]. As discussed in section 2, simply letting two relationships which have a common individual be the neighboring nodes in the CRF will not always be appropriate. Consequently, we resort to using the community structure of the social network to assist in constructing the CRF. For maintaining the structural integrity of the social network, we detect its communities over the entire dataset, rather than on the training and test subsets respectively. After the construction of the CRF, the pseudolikelihood model is trained and tested on the training and test subsets respectively. The detailed steps of our approach are as follows.

### 3.1 Step 1: Community Detection

We first run a community detection algorithm on the graph $G$ of the social network instantiation $\mathcal{I}$. According to whether intersecting communities are

generated, community detection algorithms can be divided into non-overlapping and overlapping algorithms. *Non-overlapping community detection* supposes that every individual only belongs to a single community, while *overlapping community detection* considers the natural phenomenon that one person may belong to multiple groups in real world, thus allows an individual to belong to more than one communities. Many sophisticated community detection algorithms have been developed in recent years and Fortunato [17] provides a detailed summary.

People can select non-overlapping or overlapping community detection algorithms according to the overlapping property of a social network, i.e., if an individual can belong to multiple communities. In this paper, for comparing the performance of different community detection algorithms, we use both non-overlapping and overlapping algorithms. It is noted that, a community detection algorithm is needed to be executed only once for a social network instantiation, no matter how to split the training and test subsets subsequently.

### 3.2 Step 2: Conditional Random Field Construction

We construct the CRF based on both the original social network and the community results obtained in Step 1. The principle is very simple: we treat the relationships in the original social network as the nodes in the CRF, and then establish a link between any pair of relationships if they start from the same individual and terminate in the same community. Fig. 2 lists all the possible community structures of any two neighboring relationships, in which (c) and (f) are overlapping communities. Concretely, we establish a link between any pair of relationships with a community structure like (a), (b), or (c) in the figure, while the situations like (d), (e), and (f) are ignored. Finally, we add the content attributes into the CRF as the evidence for each relationship.

We use $\mathcal{F} = (\mathbf{Y}, \mathbf{x}, \mathbf{r})$ to denote the CRF of the instantiation $\mathcal{I}$, where $\mathbf{Y}$ is the set of label variables and $\mathbf{x}$ is the set of content attributes and $\mathbf{r}$ is the set of links between the relationships. After the construction of the CRF, the Markov blanket of each label variable is determined.

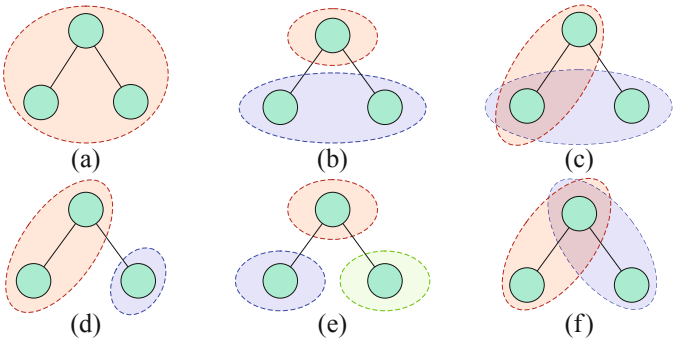


**Fig. 2.** All the possible community structures of two neighboring relationships

### 3.3 Step 3: The Pseudolikelihood Model

Given the CRF $\mathcal{F} = (\mathbf{Y}, \mathbf{x}, \mathbf{r})$, for each label $Y_i$, pseudolikelihood models use a local CPD $P(y_i|MB(y_i))$ to represent the conditional probability of the label value $y_i$, where $MB(y_i)$ is the state of the Markov blanket of $Y_i$ in the data. It is noted that, for simplifying the representation, we let the Markov blanket $MB(Y_i)$ contain not only the neighboring label variables but also the content attributes of $Y_i$. We maximize the following pseudolikelihood

$$P(\mathbf{y}|\mathbf{x}, \mathbf{r}) = \prod_{i=1}^{n} P(y_i|MB(y_i)), \tag{1}$$

where $n$ is the number of label variables in $\mathcal{F}$.

Let each pair of neighboring nodes in $\mathcal{F}$ to be a *clique* with a potential $\phi$, according to the fundamental theorem of Markov random fields [18], the conditional probability $P(y_i|MB(y_i))$ can be factorized over all of the cliques:

$$P(y_i|MB(y_i)) = \frac{1}{Z_i(\mathbf{x}_i, \mathbf{r}_i)} \prod_{v_j \in MB(y_i)} \phi(y_i, v_j), \tag{2}$$

where $Z_i$ is the local partition function (or normalization constant) given by

$$Z_i(\mathbf{x}_i, \mathbf{r}_i) = \sum_{y_i'} \prod_{v_j \in MB(y_i')} \phi(y_i', v_j). \tag{3}$$

Therefore, computing pseudolikelihood is very efficient because the local partition function is simply a sum over a single variable.

The potential is often represented by a log linear combination of a set of features:

$$\begin{aligned} \phi(y_i, v_j) &= \exp\{\sum_k w_k f_k(y_i, v_j)\} \\ &= \exp\{\mathbf{w} \cdot \mathbf{f}(y_i, v_j)\}, \end{aligned} \tag{4}$$

where $w_k$ is the weight of the feature $f_k$.

**Parameter Learning.** The goal of parameter learning is to determine the weights of the features in the pseudolikelihood model. Maximum a posterior (MAP) training is used to learn the pseudolikelihood model. To avoid overfitting, we assume the prior of the weights $\mathbf{w}$ is a zero-mean Gaussian. The log of the MAP objective function is as follows:

$$\begin{aligned} PL(\mathcal{I}, \mathbf{w}) &= \log\Big(P(\mathbf{y}|\mathbf{x}, \mathbf{r}) \prod_k P(w_k)\Big) \\ &= \log P(\mathbf{y}|\mathbf{x}, \mathbf{r}) + \sum_k \frac{-w_k^2}{2\sigma^2} - \log\sqrt{2\pi\sigma^2} \\ &= \sum_{i=1}^{n} \Big( \sum_{v_j \in MB(y_i)} \mathbf{w} \cdot \mathbf{f}(y_i, v_j) - \log Z_i \Big) - \frac{||\mathbf{w}||_2^2}{2\sigma^2} + C. \end{aligned} \tag{5}$$

$PL(\mathcal{I}, \mathbf{w})$ is a concave function and we can estimate the parameters $\mathbf{w}$ by maximizing the log-pseudolikelihood by using a variety of gradient-based optimization algorithms, such as conjugate gradient or quasi-Newton. For computing the gradient, the derivative of $PL(\mathcal{I}, \mathbf{w})$ with respect to $w_m$ is given by

$$\frac{\partial PL(\mathcal{I}, \mathbf{w})}{\partial w_k} = \sum_{i=1}^{n} \sum_{v_j \in MB(y_i)} \Big\{ f_k(y_i, v_j) - E_{P_\mathbf{w}} \big[ f_k(y_i, v_j) \big] \Big\} - \frac{w_k}{\sigma^2}, \tag{6}$$

where the expected feature values is related to $P_\mathbf{w}$:

$$E_{P_\mathbf{w}} \big[ f_k(y_i, v_j) \big] = \sum_{y_i'} \Big\{ f_k(y_i', v_j) P_\mathbf{w}(y_i', v_j) \Big\}. \tag{7}$$

The time complexity of computing the gradient in equation (6) is $O(n*r)$, where $n$ is the number of label variables and $r$ is the number of links in the CRF $\mathcal{F}$. Comparatively, the complexity of learning an RMN model is much higher because approximate inference is required, and generally the complexities of approximate inference algorithms are very high. For example, the complexity of loopy belief propagation is $O(m*n*r^2)$, where $m$ is the number of iterations. Furthermore, the use of community structure also reduces some computational cost, since the removal of some unreasonable links among the relationships makes the CRF a little sparser. In conclusion, our CBPL model is much more efficient then the RMN model in terms of computational complexity.

**Inference.** Given the observed content attributes $\mathbf{x}$ and the parameters $\mathbf{w}$, the task of inference is to determine the relationship labels. As we know, loopy belief propagation (LBP) [19,20] is often used to inference CRFs. Our inference algorithm is very similar to LBP, which estimates the marginal distribution of each label variable approximately by sending local messages through the graph structure of the model. We initialize the marginals and values of the label variables by using only the content attributes, and then update them iteratively with the state of their Markov blankets at the previous time, until each of the variables does not change any more. The detailed procedures of the inference algorithm are presented in Algorithm 1.

## 4 Experiments

In this section, we present a set of experiments to evaluate our CBPL approach. We performed relationship labeling on two real-world datasets, and compared the performance of our approach with that of the RMN model. The results of the content-only relationship labeling were taken as our baseline.

### 4.1 Datasets

**TerroristRel**[1]**.** It is a public dataset contributed by Zhao et al. [2] and collected from the PIT knowledge base. The dataset contains information about terrorists

[1] http://www.cs.umd.edu/projects/linqs/projects/lbc/

**Algorithm 1.** CBPL-Inference

**Input**: content attributes $\mathbf{x}$, links $\mathbf{r}$, parameters $\mathbf{w}$
**Output**: labels $\mathbf{Y}$

// initiation:
**foreach** *label variable* $Y_i$ **do**
  **foreach** *value* $y_i$ **do**
    // initialize the local CPD by using only the content attributes
    $P^{(0)}(y_i|MB(y_i)) \leftarrow \prod_{x_j \in MB(y_i)} \phi(y_i, x_j)/Z_i(\mathbf{x}_i, \mathbf{r}_i)$;
  **end**
  $Y_i^{(0)} \leftarrow \arg\max_{y_i} P^{(0)}(y_i|MB(y_i))$;
**end**
// iteration:
**repeat**
  **foreach** *label variable* $Y_i$ **do**
    **foreach** *value* $y_i$ **do**
      // update the local CPD by using the state of $MB(y_i)$ at $t$-1
      $P^{(t)}(y_i|MB(y_i)) \leftarrow \prod_{v_j^{(t-1)} \in MB(y_i)} \phi(y_i, v_j^{(t-1)})/Z_i(\mathbf{x}_i, \mathbf{r}_i)$;
    **end**
    $Y_i^{(t)} \leftarrow \arg\max_{y_i} P^{(t)}(y_i|MB(y_i))$;
  **end**
**until** *each variable* $Y_i$ *satisfies* $Y_i^{(t)} = Y_i^{(t-1)}$ ;

and their relationships and was designed for labeling the relationships between the terrorists. It consists of 244 terrorists and 840 relationships between them. Each relationship is described by a 0/1-valued vector where each component indicates the absence/presence of one of the total of 1224 distinct features. Each relationship can be assigned one or more labels within the labels of Family (16.0%), Colleague (54.2%), Congregate (12.4%), and Contact (17.4%).

**PhoneCallNet.** We collected a phone call network from the encrypted CDRs of the mobile users in an area in northern China obtained from one of the largest mobile service providers in China. The CDRs recorded all the phone call and short message (SM) events occurred within about 15 days in July 2010. We extracted 1623 mobile users and 4295 distinct relationships between them. For each relationship, we derived 9 statistical properties (as listed in table 1) from the CDRs and took them as the content attributes. All these statistical properties were normalized by being divided by the total call count, the total call duration, and the total SM count, respectively.

Manually labeling the relationships in this dataset for testing was not an easy task. We used the service packages provided by the mobile service provider to label the relationships. Actually, all the instances in our dataset were collected among the users who ordered at least one family or group package. Then the relationships between the users who ordered the same family package were labeled with Family (22.0%), and the relationships between the users who ordered the same group package were labeled with Group (63.3%), and the remaining relationships were labeled with Common (14.7%).

**Table 1.** The statistical properties of the relationships in the PhoneCallNet dataset

| Feature | Description |
|---|---|
| call_busy_count | the call count between 08:30 and 17:30 h on weekdays |
| call_free_count | the call count between 17:30 and 08:30 h on weekdays |
| call_weekend_count | the call count on weekend |
| call_busy_duration | the call duration between 08:30 and 17:30 h on weekdays |
| call_free_duration | the call duration between 17:30 and 08:30 h on weekdays |
| call_weekend_duration | the call duration on weekend |
| sm_busy_count | the SM count between 08:30 and 17:30 h on weekdays |
| sm_free_count | the SM count between 17:30 and 08:30 h on weekdays |
| sm_weekend_count | the SM count on weekend |

### 4.2 Experimental Setup

**Baseline.** Our approach is a link-based classification method in the relational learning field, so we focused on the comparison of our method with a representational relational learning model (i.e., RMNs). However, for achieving more information, we take the results of the content-only relationship labeling as our baseline. Traditional classifiers, such as naïve Bayes, logistic regression or SVM, can be used to perform content-only relationship labeling. In our experiments, we chose to use the conditional Markov Networks [3] in the flat setting as a representative.

**Community Detection Algorithms.** In our CBPL approach, we respectively employed the Infomap algorithm[2] [21] as the non-overlapping community detection algorithm and the Greedy Clique Expansion (GCE) algorithm[3] [22] as the overlapping one. The Infomap algorithm, proposed by Rosvall and Bergstrom, finds the best cluster structure of a graph by optimally compressing the information describing the probability flow of random walk and has a complexity that is essentially linear in the size of the graph (i.e., $O(|E|)$). It is considered as one of the best community detection algorithms so far [23]. The GCE algorithm is one of the newest overlapping community detection algorithms to detect the intersecting communities in social networks. It identifies distinct cliques as seeds and expands these seeds by greedily optimizing a local community fitness function, and then finally accepts only those communities that are not near-duplicates of communities that have already been accepted.

**Relational Clique Templates.** For the RMN model, we defined two relational clique templates as follows: 1) dyad cliques for any pair of relationships which have a common individual; and 2) triad cliques for any triple of relationships which connected end to end.

[2] http://www.tp.umu.se/~rosvall/code.html
[3] http://sites.google.com/site/greedycliqueexpansion/

**Feature Functions.** Feature functions are needed to be defined for both our CBPL approach and the RMN model. All the cliques can be divided into two categories: the cliques which consist of one label variable and one content attribute belong to the category of *evidence cliques*, while the cliques which contain only label variables belong to the category of *compatibility cliques*. For the evidence cliques, we defined a feature with the form of $f(y_i, x_j) = y_i x_j$, where $y_i = \pm 1$, and $x_j \in \{1, 0\}$ for the TerroristRel dataset and $x_j \in [0, 1]$ for the PhoneCallNet dataset. For a compatibility clique, we simply use a single feature to track whether all of its labels are the same.

**Multi-label Classification.** For the TerroristRel dataset, since two terrorists can be related in multiple relationships, multi-label classification was considered. We used a simple way that learned and tested a binary (one-against-rest) classifier for each of the four labels respectively, and then computed their average performance.

**Dataset Splitting.** A problem we often encounter in within-network learning is that directly splitting a closely connected network into separate training and test subsets would lose a lot of links which go from one subset to another. Consequently, we split our datasets according to the relationships rather than the individuals. Specifically, we put each relationship into the training or test subset with a certain probability, and in this case an individual might be in both the two subsets simultaneously. For evaluating the performance of our approach in different proportions of the observed labels, we randomly chose 10%, 20%, 30%, 40%, and 50% relationships, respectively, as the observed data for training, and the remaining relationships were used for testing.

### 4.3 Results and Discussions

The relationship labeling accuracies of the various approaches for the two datasets are shown in Fig. 3 and Fig. 4 respectively. Each experiment in this study was repeated 10 times and the results were averaged. From the two figures we can see:

1) The prediction accuracies of the relational approaches (whether the RMN model or our CBPL approach) are much better than that of the content-only approach. This demonstrates that the link structure is a very important source of information, and the relational approaches are able to learning social networks effectively by integrating information from content attributes of individuals as well as the links between them.
2) Our CBPL approach outperforms the RMN model (increases the labeling accuracies by around 2-4% for the TerroristRel dataset and 3-5% for the PhoneCallNet dataset respectively). And this demonstrates that the community structure is really very helpful for relationship labeling in social networks. Although the pseudolikelihood technique ignores the indirect dependencies and may lose some accuracy, the use of community structures makes up for this deficiency to a large extent by describing the local direct dependencies more reasonable and more accurate.

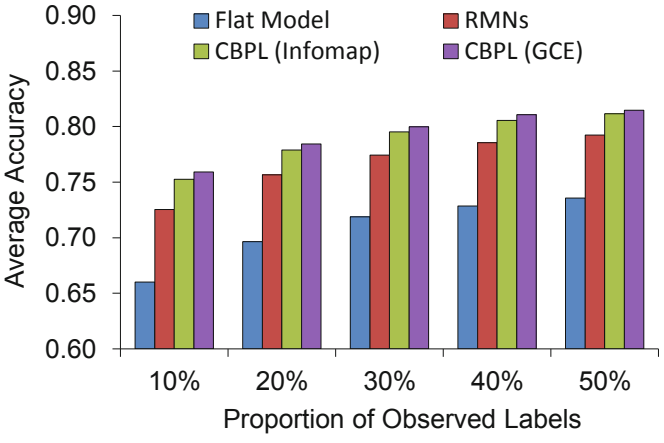


**Fig. 3.** Average classification accuracies for the TerroristRel dataset

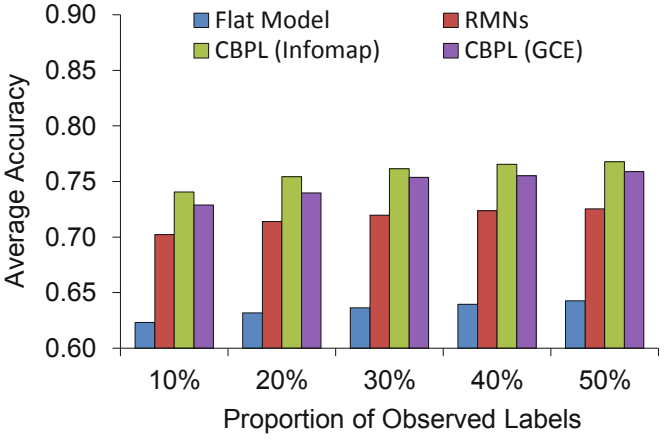


**Fig. 4.** Average prediction accuracies for the PhoneCallNet dataset

3) For the TerroristRel dataset, the CBPL approach based on overlapping community detection slightly outperforms the one based on non-overlapping community detection. The situation is just the opposite for the PhoneCallNet dataset. We believe this is due to the different community structure properties of the two datasets. That is, the overlapping nature of the communities in the TerroristRel dataset is quite strong so the CBPL approach based on the GCE algorithm performs better, and on the contrary, the overlapping nature of the PhoneCallNet dataset is weak so the CBPL approach based on the Infomap algorithm performs better.
4) Because the number of content attributes of the PhoneCallNet dataset is fewer (just 9 statistical properties), the increases of the labeling accuracies

along with the proportion of observed labels are not obvious for the various approaches. Therefore, if more features about the relationships between the communicated users were observed, the prediction accuracies could be higher.

The average training times of the various approaches along the proportion of observed labels for the two datasets are shown in Table 2 and Table 3. From the tables we see that: 1) our CBPL approach, whose training speeds are almost of the same order of magnitude as those of the flat model, is much more efficient than RMNs; and 2) the increasing rates of the training times of RMNs become much higher along with the growth of the proportion of observed labels, while those of CBPL are almost nearly linear.

**Table 2.** Average Training Times (Seconds) for the TerroristRel Dataset. All the results were computed on a PC with CPU 3.0 GHz and 2 GB RAM. Note that the time of community detection was not contained in the training times of our CBPL approach, since it is only in several milliseconds and very short comparing with the time of learning the pseudolikelihood model.

| Approach | Proportion of Observed Labels | | | | |
|---|---|---|---|---|---|
| | 10% | 20% | 30% | 40% | 50% |
| Flat Model | 0.81 | 2.06 | 3.64 | 7.79 | 11.83 |
| RMNs | 4.49 | 25.86 | 96.41 | 289.05 | 820.60 |
| CBPL (Infomap) | 2.01 | 6.02 | 15.48 | 34.85 | 51.27 |
| CBPL (GCE) | 2.53 | 7.91 | 18.96 | 39.58 | 58.73 |

**Table 3.** Average Training Times (Seconds) for the PhoneCallNet Dataset

| Approach | Proportion of Observed Labels | | | | |
|---|---|---|---|---|---|
| | 10% | 20% | 30% | 40% | 50% |
| Flat Model | 0.79 | 1.64 | 2.45 | 3.31 | 5.87 |
| RMNs | 6.53 | 33.62 | 133.84 | 437.76 | 1362.54 |
| CBPL (Infomap) | 1.26 | 4.92 | 12.85 | 27.41 | 46.05 |
| CBPL (GCE) | 1.67 | 5.63 | 15.39 | 32.27 | 53.72 |

## 5 Conclusion and Future Work

In this paper we studied the problem of relationship labeling in social networks and proposed a community-based pseudolikelihood approach. In our approach we use the community structure, one of the most important properties of complex networks, to assist us in constructing the conditional random field and the pseudolikelihood measure is employed to approximate the joint probability distribution of a collection of relationship label variables. The use of community structures makes our approach more reasonable and more accurate to describe

the dependencies between the variables in relational data, while the pseudolikelihood technique effectively resolves time complexity problem which the RMN model is suffering and makes our approach much easier to deploy in large-scale social networks.

We applied our CBPL approach to the task of relationship labeling on some real-world social networks, including a terrorist relation network and a mobile user phone call network. The experimental results showed that our approach performs well in terms of accuracy and efficiency.

There are still some works can be improved in the future. Firstly, this paper first proposes using community structure to improve link-based classification and the experiments show that the community information is really useful in practice, but the quantification of the community information should be researched further. Secondly, since our proposed approach is a supervised learning technique, fully-labeled data is needed for training the model and we must split entire social networks into separate training and test subsets in practice. Actually, semi-supervised learning may be more suitable for such partially labeled within-networks. Therefore, the development of semi-supervised community-based relationship labeling methods will be one of our future research topics. Lastly, this paper is focused on the relationship labeling problem of ordinary social networks which consist of individuals as well as the relationships between them. The generalization of our approach to multipartite or even multimode networks could also be one of our future works.

**Acknowledgments.** The authors would like to thank the anonymous reviewers for their valuable comments and suggestions. This work was supported in part by National Nature Science Foundation of China (Grant No. 60905029), Beijing Natural Science Foundation (Grant No. 4112046) and the Fundamental Research Funds for the Central Universities of China.

## References

1. Taskar, B., Wong, M.F., Abbeel, P., Koller, D.: Link prediction in relational data. In: Neural Information Processing Systems 2003, pp. 659–666. The MIT Press, Cambridge (2004)
2. Zhao, B., Sen, P., Getoor, L.: Entity and relationship labeling in affiliation networks. In: ICML 2006 Workshop on Statistical Network Analysis: Models, Issues, and New Directions (2006)
3. Taskar, B., Abbeel, B., Koller, D.: Discriminative probabilistic models for relational data. In: 18th Conference on Uncertainty in Artificial Intelligence, pp. 485–492. Morgan Kaufmann, San Francisco (2002)
4. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. PNAS 99(12), 7821–7826 (2002)
5. Besag, J.: Statistical analysis of non-lattice data. The Statistician 24(3), 179–195 (1975)
6. Getoor, L., Taskar, B.: Introduction to Statistical Relational Learning. MIT Press, Cambridge (2007)

7. Richardson, M., Domingos, P.: Markov logic networks. Technical report, Department of Computer Science and Engineering, University of Washington (2004)
8. Domingos, P., Richardson, M.: Markov logic: a unifying framework for statistical relational learning. In: ICML 2004 Workshop on Statistical Relational Learning and its Connections to Other Fields, pp. 49–54. IMLS, Washington, DC (2004)
9. Neville, J., Jensen, D.: Collective classification with relational dependency networks. In: KDD 2003 Workshop on Multi-Relational Data Mining, pp. 77–91 (2003)
10. Xiang, R., Neville, J.: Pseudolikelihood EM for within-network relational learning. In: 8th IEEE International Conference on Data Mining, pp. 1103–1108. IEEE Computer Society, Washington, DC (2008)
11. Neville, J., Jensen, D.: Leveraging relational autocorrelation with latent group models. In: 5th IEEE International Conference on Data Mining, pp. 322–329. IEEE Computer Society, Washington, DC (2005)
12. Tang, L., Liu, H.: Relational learning via latent social dimensions. In: 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 817–826. ACM Press, New York (2009)
13. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Physical Review E 69(2), 026113 (2004)
14. Wang, C., Han, J., Jia, Y., Tang, J., Zhang, D., Yu, Y., Guo, J.: Mining advisor-advisee relationships from research publication networks. In: 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 203–212. ACM Press, New York (2010)
15. Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: 18th International Conference on Machine Learning, pp. 282–289. Morgan Kaufmann, San Francisco (2001)
16. Ng, A.Y., Jordan, M.I.: On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes. In: Neural Information Processing Systems 2001, pp. 841–848. The MIT Press, Cambridge (2002)
17. Fortunato, S.: Community detection in graphs. Physics Reports 486(3-5), 75–174 (2010)
18. Kindermann, R., Snell, J.L.: Markov Random Fields and Their Applications. American Mathematical Society, Providence (1980)
19. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, San Francisco (1988)
20. Murphy, K.P., Weiss, Y., Jordan, M.I.: Loopy belief propagation for approximate inference: an empirical study. In: 15th Conference on Uncertainty in Artificial Intelligence, pp. 485–492. Morgan Kaufmann, San Francisco (1999)
21. Rosvall, M., Bergstrom, C.T.: Maps of random walks on complex networks reveal community structure. PNAS 105(4), 1118–1123 (2008)
22. Lee, C., Reid, F., McDaid, A., Hurley, N.: Detecting highly overlapping community structure by greedy clique expansion. In: KDD 2010 Workshop on Social Network Mining and Analysis (2010)
23. Lancichinetti, A., Fortunato, S.: Community detection algorithms: a comparative analysis. Physical Review E 80(5), 056117 (2009)