Within-Network Classification Using Local Structure Similarity

Christian Desrosiers and George Karypis

Department of Computer Science & Engineering,
University of Minnesota, Twin Cities
{desros,karypis}@cs.umn.edu

Abstract. Within-network classification, where the goal is to classify the nodes of a partly labeled network, is a semi-supervised learning problem that has applications in several important domains like image processing, the classification of documents, and the detection of malicious activities. While most methods for this problem infer the missing labels collectively based on the hypothesis that linked or nearby nodes are likely to have the same labels, there are many types of networks for which this assumption fails, e.g., molecular graphs, trading networks, etc. In this paper, we present a collective classification method, based on relaxation labeling, that classifies entities of a network using their local structure. This method uses a marginalized similarity kernel that compares the local structure of two nodes with random walks in the network. Through experimentation on different datasets, we show our method to be more accurate than several state-of-the-art approaches for this problem.

Keywords: Network, semi-supervised learning, random walk.

1 Introduction

Networked data is commonly used to model the relations between the entities of a system, such as hyperlinks connecting web pages, citations relating research papers, and calls between telephone accounts. In such models, entities are represented by nodes whose label gives their type, and edges are relations between these entities. As is it often the case, important information on the nature of certain entities and links may be missing from the network. The task of recovering the missing types of entities and links (i.e. node and edge labels) based on the available information, known as within-network classification, is a semi-supervised learning problem key to several applications like image processing, classifying document and web pages, classifying protein interaction and gene expression data, part-of-speech tagging, detecting malicious or fraudulent activities, and recommending items to consumers.

Classification methods for this problem suffer from two important limitations. First, many of these methods are based on the principle that nodes that are close to each other in the network are likely to have the same type, a principle known as *homophily.* While evidence suggests that homophily applies to certain

W. Buntine et al. (Eds.): ECML PKDD 2009, Part I, LNAI 5781, pp. 260–275, 2009.

networks, such as social networks [1], there are many types of networks for which this principle fails. For instance, in molecules, nearby atoms are no more likely to have the same type than distant ones. In such networks, the type of a node is instead dictated by underlying rules which may be learned by considering the relations of this node with other ones. Also, while other methods classify nodes based on their neighborhood these methods consider the distribution of labels in this neighborhood but not its structure. As we will see, the local structure of a node in the network contains important information that can improve its classification.

1.1 Contributions

This paper makes two contributions to the problem of within-network classification. First, it introduces a novel collective classification framework that combines the iterative approach of relaxation labeling with the power of similarity kernels. Unlike existing relational classifiers, which only consider the distribution of labels in the neighborhood of a node, this framework allows to use complex similarity measures between nodes. Secondly, while methods based on random walks have recently been proposed for the within-network classification problem [4,8,20], such methods evaluate the similarity between nodes using their proximity in the network. Following the success of structural kernels on the problem of graph classification [3,7,12], we present a new similarity measure inspired by marginalized graph kernels [10], that evaluates the local structure similarity between two nodes with random walks. This similarity measure upgrades marginalized kernels by considering the uncertainty of labels and the degree of nodes in the network.

The rest of this paper is organized as follows. In Section 2, we present an overview of existing methods for within-network classification. We then describe our method in Section 3, and evaluate it experimentally on several datasets in Section 4. Finally, we conclude with a short summary of our approach and contributions.

2 Related Work

Unlike traditional machine learning approaches, methods for the classification of networked data must deal with additional challenges that result from the interdependence of node classes. To overcome this problem, it has been recognized that the type of the nodes should be inferred simultaneously instead of individually, a technique known as collective classification [15]. Collective inference methods for the within-network classification problem can generally be divided in two groups: exact and approximate inference methods. Exact inference methods for this problem focus on learning the joint probability distribution of node labels. Among the best known methods of this group are those using Markov Random Fields (MRF) [11], where the joint distribution is defined as the product of potential functions that operate on the cliques of the network. Various extensions to MRFs, that also take into account observed attribute data, have also been developed. Among these are Conditional Random Fields [11], Relational Markov Networks [18] and Markov Logic Networks [6].

Due to the computational complexity of exact inference, approximation methods, such as the one presented in this paper, are normally used for the within-network classification problem. Among these is Gibbs sampling [9], which approximates the joint distribution by generating samples for the unknown labels. The main drawback of this method is its slow convergence, especially for large networks [2]. Related approaches are Relaxation Labeling (RL) [5] and Loopy Belief Propagation [19], where a vector containing the label probabilities of each node of unknown label is maintained. In RL, these vectors are initialized with apriori probabilities, either given or obtained from the data, and, at each subsequent iteration, are recomputed using a given relational classifier, until convergence or a maximum number of iterations is reached. Nodes of unknown label are then given the label of greatest probability. Unlike RL methods, Iterative Classification (IC) approaches [13,16] assign, at every iteration, a label to each node of unknown label, using a given relational classifier. To facilitate convergence, the amount of classified nodes at each iteration can be gradually increased during the process. Although our classification approach could also be used within an IC framework, we have found the updated label probabilities of RL to have better convergence properties.

2.1 Relational Classifiers

As pointed out in [15], the performance of iterative classification methods, such as RL and IC, greatly depends on the relational classifier used. A classifier strongly based on homophily, the Weighted-Vote Relational Neighbor (WVRN) [14], computes the label probability of a node as a weighted sum of the probabilities of neighbor nodes of having the same label. This simple classifier was found to work well with a RL method in the classification of documents and web pages.

Another classifier is the Class-Distribution Relational Neighbor (CDRN) [15]. This classifier assigns to each node v of known label a vector whose k-th element contain the sum, over each neighbor u of v, of the probability of u to have label k. A reference vector is then obtained for each label k as the average of the vectors belonging to nodes with known label k, and the probability of a node to have label k is defined as the similarity (L_1, L_2, cosine, etc.) between its vector and the reference vector of label k. While our classification approach is also based on node similarity, it is more general than CDRN which only considers the distribution of labels in the neighborhood of a node.

Two other relational classifiers are the Network-Only Bayes (NOB) classifier [5] and the Network-Only Link-Based (NOLB) [13] classifier. The former, which was originally used with an RL method to classify documents employs a naive Bayes approach to compute the label probability of a node, assumed to be independent given the labels of its neighbors. Finally, the NOLB classifier learns a multiclass logistic regression model using the label distribution (raw or normalized counts, or aggregation of these values) in the neighborhood of nodes with known labels. In [13], this classifier was used within an IC method to classify documents. As with CDRN, these methods do not use the local structure of a node in its classification.

3 A Novel Classification Approach

Our method is composed of two parts: a classification approach based on relaxation labeling and a node structure similarity inspired by marginalized kernels.

3.1 Relaxation Labeling Framework

Although the methods presented in this paper could be extended to the multivariate case of the within-network classification problem, we will focus on the univariate case.

We model relational data as a partially labeled graph $G = (V, E, W, L_V, L_E, l)$ where V is a set of nodes, E a set of edges between the nodes of V, $W \subset V$ is the set of nodes for which the true labels are known, L_V and L_E are respectively the sets of node and edge labels, and l is a function that maps each node and edge to a label of the corresponding set. We denote by l_u the label of a node u and $l_{u,v}$ the label of an edge (u, v). Denoting U the set of unlabeled nodes of G, i.e. $U = V \setminus W$, the problem consists in assigning to each $u \in U$ a label in L_V based on the labels of nodes in W.

As with other RL methods, our approach works by iteratively updating the label probabilities of each unlabeled node using a relational classifier, until convergence. Let K_u be a random variable modeling the label of a node u. Our relational classifier is based on the assumption that the probability $\mathrm{P}(K_u = k)$ of a node u to have label k is determined by the membership of u to a certain "node class", modeled by random variable C_u whose possible values are the nodes of V. Thus, $\mathrm{P}(C_u = v)$ represents the probability of node u to "behave" in the same way as node v, or more simply, the similarity between u and v. Following this model, $\mathrm{P}(K_u = k)$ can be obtained by marginalizing C_u:

$$\begin{aligned}\mathrm{P}(K_u = k) &= \sum_{v \in V} \mathrm{P}(K_u = k, C_u = v) \\ &= \sum_{v \in V} \mathrm{P}(K_u = k | C_u = v)\, \mathrm{P}(C_u = v) \\ &\propto \sum_{v \in V} \mathrm{P}(K_v = k)\, \mathrm{P}(C_u = v).\end{aligned}$$

Using the shorthand notation $\pi_{v,k} = \mathrm{P}(K_v = k)$ and letting $\sigma_{u,v} \propto \mathrm{P}(C_u = v)$ denote the similarity between u and v, this expression becomes

$$\pi_{u,k} = \frac{1}{Z} \sum_{v \in V} \pi_{v,k}\, \sigma_{u,v},$$

where Z is a normalization constant. Assuming the label probabilities are all binary, i.e. $\pi_{v,k} = \delta\,(l_v = k)$, and letting $V_k \subseteq V$ be the nodes that have label k, the model simplifies to

$$\pi_{u,k} \;=\; \frac{1}{Z}\sum_{v\in V_k} \sigma_{u,v}.$$

This expression underlines an important drawback, where the probability of a label k is proportional the number of nodes that have k as label. In cases where there is an important bias in the distribution of labels, such as those reported in the experimental section, this causes all the unlabeled nodes to get the most frequent label. To alleviate this problem, we use a different formulation where

$$\pi_{u,k} \;=\; \frac{1}{Z}\cdot\frac{\sum\limits_{v\in V}\pi_{v,k}\,\sigma_{u,v}}{\sum\limits_{v\in V}\pi_{v,k}}.$$

If we consider, once again, the label probabilities as binary, this expression becomes

$$\pi_{u,k} \;=\; \frac{1}{Z}\cdot\frac{1}{|V_k|}\sum_{v\in V_k}\sigma_{u,v},$$

i.e., $\pi_{u,k}$ is proportional to the average similarity with nodes of label k.

Finally, this model is augmented with two parameters $\alpha \geq 0$ and $\beta \geq 0$ which respectively encode the importance given to label uncertainty and node similarity:

$$\pi_{u,k} \;=\; \frac{1}{Z}\cdot\frac{\sum\limits_{v\in V}\pi^{\alpha}_{v,k}\,\sigma^{\beta}_{u,v}}{\sum\limits_{v\in V}\pi^{\alpha}_{v,k}}. \qquad (1)$$

Our classification approach can be summarized with the following iterative process. First, we initialize the label probability of unlabeled nodes using apriori probabilities, either known or approximated from the labeled nodes. Then, at each iteration, we use (1) to update the label probabilities $\pi_{u,k}$ of each unlabeled node $u \in U$ and label $k \in L_V$, using the values of the previous iteration. These values are then normalized to make sure that they constitute a probability distribution. This process is repeated until the label probabilities converge, i.e. the average change is inferior to a given threshold $\epsilon > 0$, or we reach a given number of iteration $T_{\max}$. Finally, we assign to each unlabeled node $u \in U$ the label k of highest probability $\pi_{u,k}$.

Note that this approach can also be used to classify the unlabeled edges of a graph G. The idea is to transform G by replacing each edge $(u, v) \in E$ by a new node uv and two new edges, (u, uv) and (uv, v). The graph obtained in this way has $|V| + |E|$ nodes with labels from a set of $|L_V| + |L_E|$ nodes labels, and $2|E|$ edges with the same label.

3.2 Random Walk Structure Similarity

Our approach to evaluate the local structure similarity of two nodes is based on marginalized graph kernels [10], which compute similarities as the probability of generating the same sequence of labels in two parallel random walks.

While a more general approach using product graphs has been proposed to compute the structural similarity between graphs [7], the probabilistic framework of marginalized kernels is better suited to cope with the label uncertainties of our RL method. We should also mention that other types of kernels have been proposed to measure the similarity between *nodes*, such as exponential, diffusion and regularization kernels [17], and kernels based on random walks [4,8,20]. However, these kernels are mostly based on the proximity of the nodes in the graph, not their structural similarity.

Our similarity measure differs from marginalized kernels in two respect. First, it evaluates the similarity between two nodes of a same graph, instead of between two different graphs. Accordingly, the similarity between two nodes u and u' is defined as the probability of generating the same sequence with random walks starting at u and u'. Secondly, the labels of some nodes are only known as a probability. To cope with this problem, we make the label generation stochastic such that label k is generated at node v with probability $\pi_{v,k}$.

Since we do not demonstrate the semi-definite positiveness of our proposed kernels and because our classification framework is meant to be very flexible, the term *kernel* should be considered as a *general similarity measure* throughout the rest of the paper.

3.3 Derivation of the Similarity Kernel

Denote by $\mathrm{P}_t(v|u)$ the probability that the walk jumps from a node u to an adjacent node v and $\mathrm{P}_e(v)$ the probability that the walk stops at node v, satisfying the constraint that

$$\mathrm{P}_e(u) + \sum_{v \in V} \mathrm{P}_t(v|u) \;=\; 1. \tag{2}$$

Following these definitions, the probability of visiting a sequence of nodes $\mathbf{v} = (\mathbf{v}_0, \ldots, \mathbf{v}_n)$ in a random walk starting at node $\mathbf{v}_0$ is

$$\mathrm{P}(\mathbf{v}) = \left(\prod_{i=1}^{n} \mathrm{P}_t(\mathbf{v}_i|\mathbf{v}_{i-1}) \right) \mathrm{P}_e(\mathbf{v}_i).$$

Let $\mathrm{P}_l(k|v)$ and $\mathrm{P}_l(k|u,v)$ denote, respectively, the probability of generating label $k \in L_V$ at node v and the probability of generating label $k' \in L_E$ while traversing edge (u,v). Given node sequence $\mathbf{v}$, the conditional probabilities of generating the sequences of node labels $\mathbf{s}$ and edge labels $\mathbf{q}$ are

$$\begin{aligned} \mathrm{P}(\mathbf{s}|\mathbf{v}) \;&=\; \prod_{i=1}^{n} \mathrm{P}_l(\mathbf{s}_i|\mathbf{v}_i) \\ \mathrm{P}(\mathbf{q}|\mathbf{v}) \;&=\; \prod_{i=1}^{n} \mathrm{P}_l(\mathbf{q}_i|\mathbf{v}_{i-1}, \mathbf{v}_i) \end{aligned}$$

Let $\mathcal{W}_u^{(n)}$ be the set of possible sequences of $n+1$ nodes visited in a random walk starting at node u. The marginalized probability of a sequence $\mathbf{s}$, given a start node $u = \mathbf{v}_0$, is obtained by summing over all sequences of $\mathcal{W}_u^{(n)}$:

$$\begin{aligned}
\mathrm{P}(\mathbf{s},\mathbf{q}|u) &= \sum_{n=1}^{\infty} \sum_{\mathbf{v}\in\mathcal{W}_u^{(n)}} \mathrm{P}(\mathbf{s}|\mathbf{v})\mathrm{P}(\mathbf{q}|\mathbf{v})\mathrm{P}(\mathbf{v}) \\
&= \sum_{n=1}^{\infty} \sum_{\mathbf{v}} \left(\prod_{i=1}^{n} \mathrm{P}_t(\mathbf{v}_i|\mathbf{v}_{i-1})\mathrm{P}_l(\mathbf{s}_i|\mathbf{v}_i)\mathrm{P}_l(\mathbf{q}_i|\mathbf{v}_{i-1},\mathbf{v}_i) \right) \mathrm{P}_e(\mathbf{v}_i).
\end{aligned}$$

Denote by $\mathcal{S}^{(n)}$ and $\mathcal{Q}^{(n)}$ the set containing, respectively, all sequences of n node labels and edge labels, the probability of generating the same sequence in two parallel random walks starting at nodes u and u' is given by

$$\begin{aligned}
\sigma_{u,u'} &= \sum_{n=1}^{\infty} \sum_{\mathbf{s}\in\mathcal{S}^{(n)}} \sum_{\mathbf{q}\in\mathcal{Q}^{(n)}} \mathrm{P}(\mathbf{s},\mathbf{q}|u)\mathrm{P}(\mathbf{s},\mathbf{q}|u') \\
&= \sum_{n=1}^{\infty} \sum_{\mathbf{s},\mathbf{q}} \sum_{\mathbf{v}\in\mathcal{W}_u^{(n)}} \sum_{\mathbf{v}'\in\mathcal{W}_{u'}^{(n)}} \left(\prod_{i=1}^{n} \mathrm{P}_t(\mathbf{v}_i|\mathbf{v}_{i-1})\mathrm{P}_t(\mathbf{v}'_i|\mathbf{v}'_{i-1})\mathrm{P}_l(\mathbf{s}_i|\mathbf{v}_i)\mathrm{P}_l(\mathbf{s}_i|\mathbf{v}'_i) \right. \\
&\qquad \left. \mathrm{P}_l(\mathbf{q}_i|\mathbf{v}_{i-1},\mathbf{v}_i)\mathrm{P}_l(\mathbf{q}_i|\mathbf{v}'_{i-1},\mathbf{v}'_i) \right) \mathrm{P}_e(\mathbf{v}_n)\mathrm{P}_e(\mathbf{v}'_n) \\
&= \sum_{n=1}^{\infty} \sum_{\mathbf{s},\mathbf{q}} \sum_{\mathbf{v},\mathbf{v}'} \left(\prod_{i=1}^{n} a(\mathbf{v}_{i-1},\mathbf{v}'_{i-1},\mathbf{v}_i,\mathbf{v}'_i,\mathbf{s}_i,\mathbf{q}_i) \right) \mathrm{P}_e(\mathbf{v}_n)\mathrm{P}_e(\mathbf{v}'_n),
\end{aligned}$$

where

$$a(\mathbf{v}_{i-1},\mathbf{v}'_{i-1},\mathbf{v}_i,\mathbf{v}'_i,\mathbf{s}_i,\mathbf{q}_i) = \mathrm{P}_t(\mathbf{v}_i|\mathbf{v}_{i-1})\mathrm{P}_t(\mathbf{v}'_i|\mathbf{v}'_{i-1})\mathrm{P}_l(\mathbf{s}_i|\mathbf{v}_i)\mathrm{P}_l(\mathbf{s}_i|\mathbf{v}'_i)\mathrm{P}_l(\mathbf{q}_i|\mathbf{v}_{i-1},\mathbf{v}_i)\mathrm{P}_l(\mathbf{q}_i|\mathbf{v}'_{i-1},\mathbf{v}'_i).$$

The computation of $\sigma_{u,u'}$ can be greatly simplified using the following recurrence: the probability of generating the same sequence of n labels two parallel random walks starting at nodes u and u', written $r_{u,u'}^{(n)}$, can be obtained from the probability of visiting nodes v and v', respectively from u and u', and the probability of generating the same sequences of $n-1$ node and edge labels, starting at nodes v and v'. This recurrence can be written as

$$r_{u,u'}^{(n)} = \begin{cases} \displaystyle\sum_{v,v'\in V} \sum_{k\in L_V} \sum_{k'\in L_E} a(u,u',v,v',k,k')\; r_{v,v'}^{(n-1)} & ,n \geq 1 \\ \mathrm{P}_e(u)\; \mathrm{P}_e(u') & ,n = 0 \end{cases}$$

The probability of generating the same sequences of at most N labels starting from nodes u and u', written $R_{u,u'}^{(N)}$, is then

$$\begin{aligned}
R_{u,u'}^{(N)} &= \sum_{n=1}^{N} r_{u,u'}^{(n)} \\
&= \sum_{n=1}^{N} \sum_{v,v'\in V} \sum_{k\in L_V} \sum_{k'\in L_E} a(u,u',v,v',k,k')\; r_{v,v'}^{(n-1)} \\
&= \sum_{v,v'} \sum_{k,k'} a(u,u',v,v',k,k') \sum_{n=1}^{N} r_{v,v'}^{(n-1)} \\
&= \sum_{v,v'} \sum_{k,k'} a(u,u',v,v',k,k')\; \left(\mathrm{P}_e(v)\mathrm{P}_e(v') + R_{v,v'}^{(N-1)} \right),
\end{aligned}$$

where $R_{u,u'}^{(0)} = 0$ for all u,u'. We then have $\sigma_{u,u'} = \lim_{N\to\infty} R_{u,u'}^{(N)}$.

Denote by N_u the neighbors of node u and let $d_u = |N_u|$ be the degree of u. Setting the termination probabilities of u to a constant $\mathrm{P}_e(u) = \gamma$, and letting the transition probabilities be uniform over the neighbors of u, following the constraint of (2), we have $\mathrm{P}_t(v|u) = (1-\gamma)/d_u$ if $v \in N_u$ and 0 otherwise. Furthermore, using $\mathrm{P}_l(k|v) = \pi_{v,k}$ and $\mathrm{P}_l(k'|u,v) = \delta\left(l_{u,v} = k'\right)$ as node and edge label probabilities, the formulation of the kernel becomes

$$R^{(N)}_{u,u'} = \frac{(1-\gamma)^2}{d_u d_{u'}} \sum_{v \in N_u} \sum_{v' \in N_{u'}} \sum_{k \in L_V} \delta\left(l_{u,v} = l_{u',v'}\right) \pi_{v,k} \pi_{v',k} \left(\gamma^2 + R^{(N-1)}_{v,v'}\right). \quad (3)$$

Other than being computed between pairs of nodes instead of graphs, this expression differs from the one of [10] by the fact that the label probabilities are also marginalized. To compute the kernel, we use a bottom-up iterative approach, where we use (3) to compute the probabilities $R^{(N)}$ based on $R^{(N-1)}$. We repeat this process for increasing values of N, until the similarity values converge, i.e. the average change is smaller than a given ϵ, or N reaches a given limit $N_{\max}$.

3.4 Exploiting Node Degrees

A problem with the kernel definition of (3) is that it does not consider the difference between the degrees of two nodes u and v, while evaluating their similarity. To illustrate this, suppose we limit the walk length in (3) to $N_{\max} = 1$, i.e. we consider only the direct neighbors of u and v. Moreover, suppose that the label of every node is known, i.e. $\pi_{u,k} = \delta\left(l_u = k\right)$. Under these constraints, the similarity kernel becomes

$$\sigma_{u,v} = \frac{(1-\gamma)^2\gamma^2}{d_u d_v} \sum_{k \in L_V} n_{u,k}\, n_{v,k},$$

where $n_{u,k} \leq d_u$ denotes the number of neighbors of u that have label k. Thus, this simplified kernel simply compares ratios of neighbors having each label k, similar to what is done in the CDRN classifier. Using this formulation, the similarity between the nodes u and v of Figure 1 (a)-(b) is equal to the self-similarity of these nodes: $\sigma_{u,u} = \sigma_{v,v} = \sigma_{u,v} = \frac{1}{2}(1-\gamma)^2\gamma^2$.

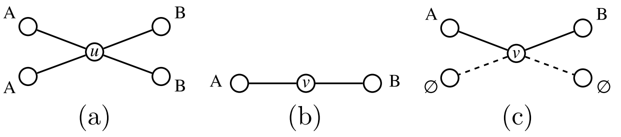

Fig. 1. (a)-(b) The neighborhood of two nodes u, v and (c) the transformed neighborhood of v

In order to consider the difference in the degrees, we modify the kernel formulation as

$$R_{u,u'}^{(N)} = \frac{(1-\gamma)^2}{\max\{d_u, d_{u'}\}^2} \sum_{v \in N_u} \sum_{v' \in N'_u} \sum_{k \in L_V} \delta\left(l_{u,v} = l_{u',v'}\right) \pi_{v,k}\pi_{v',k} \left(\gamma^2 + R_{v,v'}^{(N-1)}\right) \quad (4)$$

This modification to the kernel can be interpreted in the parallel random walks framework as follows. If the degree of the node visited by a walk is less than the degree of the node visited by the other walk, temporary edges are added from this node to a dummy node of label $\varnothing \notin L_V$, such that both nodes have the same degree. With the same probability as the true neighbors, the random walk can jump to this dummy node, after which the probability of generating the same sequence becomes null. Figure 1(c) illustrates this idea for nodes u, v of (a) and (b). Using this new formulation, the similarity values for nodes u and v, again limiting the walk length to $N_{\max} = 1$, are $\sigma_{u,u} = \sigma_{v,v} = \frac{1}{2}(1-\gamma)^2\gamma^2 \geq \frac{1}{4}(1-\gamma)^2\gamma^2 = \sigma_{u,v}$.

3.5 Convergence and Complexity

While the convergence of the similarity kernels defined above is shown in Appendix A, the collective classification method presented in this paper, as most RL methods, is not guaranteed to converge since the node structure similarities $\sigma_{u,v}$ vary from one iteration to the next. However, by limiting the number of allowed iterations to $T_{\max}$, we can still obtain a solution in the non-converging case. Furthermore, while the classification process can be expensive in the worst-case, i.e. $O\left(T_{\max}N_{\max}d_{\max}^2|L_V||V|^2\right)$, its complexity is closer to $O(|V|^2)$ in practice due to four reasons: 1) there are much less node labels than nodes, 2) the nodes of many real-life graphs have a low bounded degree (e.g., molecular graphs), 3) the relevant structural information of a node is contained within a short distance, and 4) the RL algorithm normally converges in a few iterations, regardless of $|V|$.

4 Experimental Evaluation

In this section, we test our framework on the problem of classifying the unlabeled nodes of a partly labeled graph.

4.1 Experimental Setting

We tested our classification approach on five datasets. The first three datasets, which are available online at the IAM Graph Database Repository[1], were originally used for the prediction of mutagenicity, AIDS antiviral activity, and protein function. The first two model chemical compounds as undirected graphs where the nodes represent atoms, node labels are the chemical symbols of these

[1] `http://www.iam.unibe.ch/fki/databases/iam-graph-database`

Table 1. Properties of the datasets

Property	Mutagen.	AIDS	Protein	Cornell	Texas
Nb. graphs	4,337	2,000	600	1	1
Avg. nodes	30.3	15.7	32.6	351	338
Avg. edges	30.8	16.2	62.1	1392	986
Node labels	14	38	3	6	6
Edge labels	3	3	5	1	1
Freq. class	44.3%	59.3%	49.4%	41.5%	48.1%

atoms, and edges are covalent bonds between atoms. Edge labels give the valency of these bonds. The third dataset models proteins into undirected graphs using their secondary structure, such that nodes are secondary structure elements (SSE) labeled as helix, sheet, or turn. Every node is connected with an edge to its three nearest neighbors[2] in space, and edges are labeled with their structural type.

Finally, the last two datasets, which were created for the WebKB project, contain graphs modeling the links between Web pages collected from computer science departments of the Cornell and Texas Universities. These two datasets, available online[3], have often been used to benchmark within-network classification methods, as in [15]. While the link information is sometimes converted into a co-citation graph, we evaluate our approach directly on the original Web page link graph. Furthermore, we consider the multiclass classification problem where pages can have one of six types: *student, faculty, staff, department, course* and *project*. Finally, while they are used in the evaluation of other methods, the edges weights representing the number of links between two Web pages, are ignored by our methods.

Table 1 gives some properties of these datasets: the number of graphs, the average number of nodes and edges of these graphs, their number of node and edge labels, and the percentage of nodes having the most frequent class label.

As suggested in [15], we compare our approach with the classification methods implemented in NetKit-SRL[4]. This toolkit provides a general framework for within-network classification that allows the user to choose any combination of collective inference approach, i.e. RL, IC or Gibbs sampling, and relational classifier, i.e. WVRN, CDRN, NOB or NOLB (using either raw or normalized counts of neighbors with a given label). For additional information, the reader may refer to Section 2 or to [15]. Although we have tested every possible combination of collective classification approach and relational classifier, we have kept, for each classifier, the approach which worked best. Including the two methods proposed in this paper, i.e. our RL framework with the similarity kernels of (3) and (4), a total of 7 methods, described in Table 2, are tested.

[2] Note that a node can have more than three neighbors since the relation "nearest-neighbor" is not symmetric.

[3] `http://netkit-srl.sourceforge.net/data.html`

[4] `http://netkit-srl.sourceforge.net/`.

Table 2. Tested classification methods

Method	Description
RL-WVRN:	RL with WVRN
RL-CDRN:	RL with CDRN (cosine similarity on normalized counts)
IC-NOB:	IC with NOB
IC-NOLB-count:	IC with NOLB (raw counts)
IC-NOLB-norm:	IC with NOLB (normalized counts)
RL-RW:	Our RL with the kernel of (3).
RL-RW-deg:	Our RL with the kernel of (4).

The five datasets were used differently in our experiments. For the first three ones, which contain many small graphs, we randomly sampled six sets of 100 graphs and then merged the graphs of each of these sets into larger test graphs, considering the small graphs as individual components of the larger ones (1500 to 3500 nodes depending on the dataset). We then randomly selected one of these test graphs to tune the parameters of the tested methods and used the five others to evaluate their performance. For each of these five test graphs, 10 runs were performed, where we randomly selected a subset of nodes from which we removed the labels. We then computed the F1-score using the precision and recall obtained for each class, weighted by the number of nodes in these classes, and averaged this value over the 5×10 classification runs. For the graphs of the last two datasets, parameters were tuned using another WebKB dataset modeling the links between Web pages of the University of Washington. As with the other datasets, 10 runs were performed on each of these two graphs and the F1-scores were averaged over these runs.

4.2 Results

Figures 2 gives the F1-scores obtained by the seven tested methods on the five datasets, for decreasing percentages of labeled nodes. Note that we have used a different range of labeled nodes for the two WebKB graphs (10% to 80% instead of 2.5% to 50%), since these graphs have much less nodes than the other ones.

We can see that our structure similarity kernel approach that considers node degrees, i.e. RL-RW-deg, outperforms the other classification methods for datasets where the type of a node is well correlated with its local structure (i.e., Mutagenicity and AIDS datasets), especially when a small portion of nodes are labeled. Moreover, within the classification methods of Netkit-SRL, the IC method based on the multiclass regression using the raw counts, i.e. IC-NOLB-count, provides results comparable with RL-RW-deg when the labels of a sufficient number of nodes are known. However, as the number of labeled nodes reduces, this method fails to learn a proper regression model and its performance drops. Finally, the methods based on homophily, such as RL-WDRN, perform poorly on this type of data.

Our classification approach considering node degrees also works well on other types of data, such as the Web page link graphs, where it is comparable to

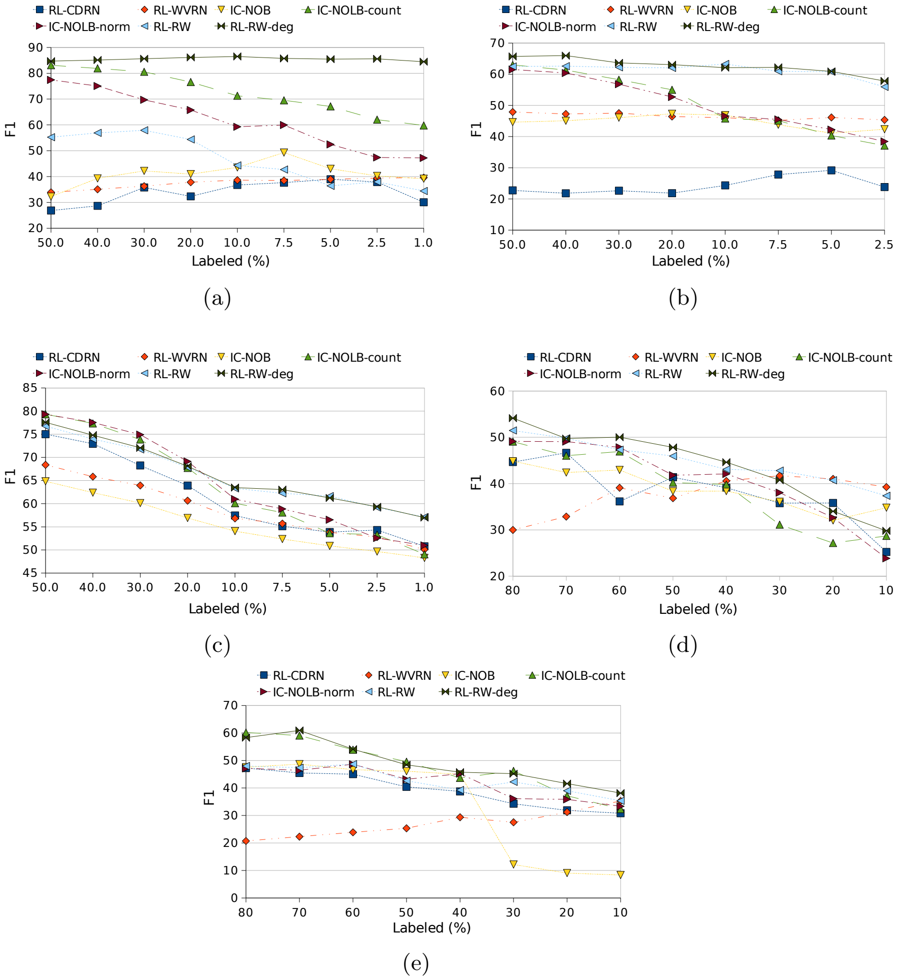

Fig. 2. F1-scores obtained for the tested datasets: (a) Mutagenicity, (b) AIDS, (c) Protein, (d) Cornell and (e) Texas

the best NetKit-SRL method. For the Cornell dataset, however, it appears that WDRN works the best when a few labels are known. In this case, this method simply assigns the most frequent label to unlabeled nodes, which gives decent results due to the biased distribution of labels (see Table 1). As more node labels are known, the classification relies increasingly on proximity which actually degrades the performance of this method.

Comparing our two similarity kernels, we observe a variation in the results obtained for the different types of data. Thus, while RL-RW-deg is significantly better than RL-RW on the Mutagenicity data, the performance of these two methods is comparable on the AIDS data. This is due to the fact valency of an atom, i.e. degree of a node, is a good indicator of the type of this atom, but this information is noisy in the AIDS data since bonds to hydrogen atoms have been omitted. For the Protein and WebKD datasets, however, the degree of a node provides a weaker signal for classification, and both approaches give comparable results.

4.3 Influence of Parameters and Runtimes

Although the results presented in this section were obtained on the same datasets as for the validation, these results were not used to tune our methods.

Figure 3(a) gives the average accuracy of RL-RW-deg on the Mutagenicity data (using a percentage of labeled nodes of 50%) for different values of parameters α and β, which control the impact of label uncertainty and similarity in our relational classifier. We notice that the accuracy can sometimes be improved by increasing the importance of nodes with uncertain labels w.r.t. nodes of known label, i.e. using $\alpha < 1$. This could be explained by the fact that using such values provides a smoother convergence of the method. This could also explain the poor results of the RL-CDRN method, which corresponds to using $\alpha \rightarrow \infty$ in our framework (assuming the random walk length is limited to 1).

The impact of the random walk termination probability γ on the classification of the AIDS data (using a percentage of labeled nodes of 50%) is shown in Figure 3(b). To illustrate how this parameter influences the length of the random walks, we varied the maximum walk length $N_{\max}$ of the kernel. When the walk lengths are the least limited, i.e. $N_{\max} = 6$, we notice that the accuracy is reduced when the termination probability γ increases. We also see that the greatest gain in accuracy occurs for $N_{\max} = 2$, suggesting that most of the structural information of a node, for this data, is contained within a short distance of this node.

The last analysis focuses on the times required to run our methods on a machine equipped with two 2.60GHz i686 processors and 1Gb of RAM. Figure 4 gives the mean runtimes of RL-RW-deg on the Mutagenicity data (using a percentage of labeled nodes of 50%), for different values of kernel parameter γ. As a reference, we also give the runtime of RL-CDRN, the slowest Netkit-SRL classification method

	RL β						
RL α	0.5	1.0	1.5	2.0	2.5	3.0	3.5
0.25	87.42	87.48	87.75	88.22	88.42	88.68	89.28
0.50	87.42	87.55	88.02	88.42	88.42	88.82	89.41
0.75	86.95	87.95	88.15	87.88	88.02	88.68	88.88
1.00	86.42	87.82	87.75	87.08	85.82	83.69	81.89
1.25	86.22	86.88	84.82	82.36	77.03	72.44	67.44
1.50	84.75	83.02	76.23	69.04	59.19	44.67	43.81

(a)

Kernel	Kernel γ				
$N_{\max}$	0.1	0.3	0.5	0.7	0.9
1	62.54	62.54	62.54	62.54	62.54
2	66.92	66.92	67.49	65.80	62.76
3	65.69	67.15	66.25	64.45	62.76
4	65.46	67.82	66.02	64.00	62.76
5	67.71	67.82	66.02	64.00	62.76
6	66.92	67.60	66.02	64.00	62.76

(b)

Fig. 3. Impact of the parameters on the classification accuracy: (a) RL parameters α and β on the Mutagenicity data, and (b) kernel parameters $N_{\max}$ and γ on the AIDS data

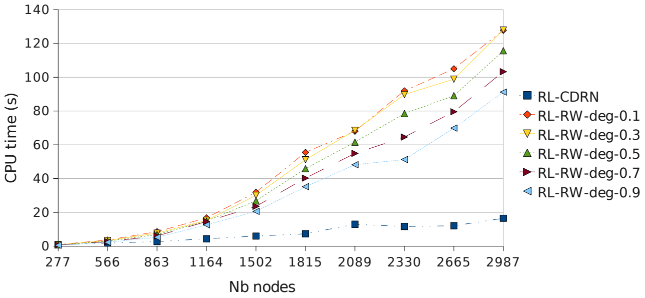

Fig. 4. Runtime (in seconds) of our approach on the Mutagenicity data

for this data. While our method is noticeably slower than methods based only on direct neighbors, such as RL-CDRN, it performed the classification within only 2 minutes for networks with 3,000 nodes, suggesting it could be used for even larger networks.

5 Conclusion

This paper presented a novel approach for the problem of within-network classification. Unlike other methods for this problem, which are based on the principle that nearby nodes in the network are likely to have the same type, or which use only the distribution of labels in the neighborhood of a node, this approach classifies a node based on its local structure similarity with other nodes in the network. Furthermore, a new method was proposed to evaluate the structural similarity between nodes in a partly labeled graph. This method, which uses random walks, extends marginalized graph kernels by considering the label uncertainty and the degree of nodes in the network. Our classification approach was tested on real-life data from the several fields, and the experimental results have shown our method to outperform several state-of-the-art methods when the type of a node is correlated to its structure, and perform as well as these methods with other types of data.

Acknowledgements. This work was supported by NSF ACI-0133464, IIS-0431135, NIH RLM008713A, and by the Digital Technology Center at the University of Minnesota.

References

1. Barabasi, A., Jeong, H., Neda, Z., Ravasz, E., Schubert, A., Vicsek, T.: Evolution of the social network of scientific collaborations. Physica A 311(3-4), 590–614 (2002)
2. Besag, J.: On the statistical analysis of dirty pictures. Journal of the Royal Statistical Society 48(3), 259–302 (1986)
3. Borgwardt, K., Ong, C., Schönauer, S., Vishwanathan, S., Smola, A., Kriegel, H.-P.: Protein function prediction via graph kernels. Bioinformatics 21(1), 47–56 (2005)

4. Callut, J., Francoisse, K., Saerens, M., Dupont, P.: Semi-supervised classification from discriminative random walks. In: Daelemans, W., Goethals, B., Morik, K. (eds.) ECML PKDD 2008, Part I. LNCS (LNAI), vol. 5211, pp. 162–177. Springer, Heidelberg (2008)
5. Chakrabarti, S., Dom, B., Indyk, P.: Enhanced hypertext categorization using hyperlinks. In: SIGMOD 1998: Proc. of the 1998 ACM SIGMOD Int. Conf. on Management of data, pp. 307–318. ACM Press, New York (1998)
6. Domingos, P., Richardson, M.: Markov logic: A unifying framework for statistical relational learning. In: Proc. of the ICML 2004 Workshop on Statistical Relational Learning and its Connections to Other Fields, pp. 49–54 (2004)
7. Gaertner, T., Flach, P., Wrobel, S.: On graph kernels: Hardness results and efficient alternatives. In: Proc. of the 16th Annual Conf. on Computational Learning Theory, August 2003, pp. 129–143. Springer, Heidelberg (2003)
8. Gallagher, B., Tong, H., Eliassi-Rad, T., Faloutsos, C.: Using ghost edges for classification in sparsely labeled networks. In: KDD 2008: Proc. of the 14th ACM SIGKDD Int. Conf. on Knowledge discovery and data mining, pp. 256–264. ACM Press, New York (2008)
9. Geman, S., Geman, D.: Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. In: Neurocomputing: foundations of research, pp. 611–634 (1988)
10. Kashima, H., Tsuda, K., Inokuchi, A.: Marginalized kernels between labeled graphs. In: Proc. of the 12th In. Conf. on Machine Learning, pp. 321–328. AAAI Press, Menlo Park (2003)
11. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: ICML 2001: Proc. of the 18th Int. Conf. on Machine Learning, pp. 282–289. Morgan Kaufmann Publishers Inc., San Francisco (2001)
12. Li, X., Zhang, Z., Chen, H., Li, J.: Graph kernel-based learning for gene function prediction from gene interaction network. In: BIBM 2007: Proc. of the 2007 IEEE Int. Conf. on Bioinformatics and Biomedicine, Washington, DC, USA, pp. 368–373. IEEE Computer Society Press, Los Alamitos (2007)
13. Lu, Q., Getoor, L.: Link-based classification. In: Fawcett, T., Mishra, N., Fawcett, T., Mishra, N. (eds.) Proc. 12th Int'l Conf. Machine Learning (ICML), pp. 496–503. AAAI Press, Menlo Park (2003)
14. Macskassy, S.A., Provost, F.: A simple relational classifier. In: Proc. of the 2nd Workshop on Multi-Relational Data Mining (MRDM 2003), pp. 64–76 (2003)
15. Macskassy, S.A., Provost, F.: Classification in networked data: A toolkit and a univariate case study. Journal of Machine Learning Research 8, 935–983 (2007)
16. Neville, J., Jensen, D.: Iterative classification in relational data. In: Proc. Workshop on Statistical Relational Learning, AAAI, pp. 13–20. AAAI Press, Menlo Park (2000)
17. Smola, A., Kondor, R.: Kernels and regularization on graphs. In: Warmuth, M., Schölkopf, B. (eds.) Proc. of the 2003 Conf. on Computational Learning Theory (COLT) and Kernels Workshop, pp. 144–158 (2003)
18. Taskar, B., Abbeel, P., Koller, D.: Discriminative probabilistic models for relational data. In: UAI 2002, Proc. of the 18th Conf. in Uncertainty in Artificial Intelligence, pp. 485–492. Morgan Kaufmann, San Francisco (2002)
19. Yedidia, J.S., Freeman, W.T., Weiss, Y.: Constructing free-energy approximations and generalized belief propagation algorithms. IEEE Transactions on Information Theory 51(7), 2282–2312 (2005)

20. Zhu, X., Ghahramani, Z., Lafferty, J.: Semi-supervised learning using gaussian fields and harmonic functions. In: Proc. of the 12th Int. Conf. on Machine Learning (ICML), pp. 912–919 (2003)

A Proof of Convergence

Proposition 1. *The kernel defined by (3) converges for any* $0 < \gamma \leq 1$.

Proof. Consider the probability of generating the same sequence of n labels in two random walks starting at u and u', as defined by the following equation:

$$r^{(n)}_{u,u'} = \frac{(1-\gamma)^2}{d_u d_{u'}} \sum_{v \in N_u} \sum_{v' \in N_{u'}} \sum_k \delta\left(l_{u,v} = l_{u',v'}\right) \pi_{v,k}\, \pi_{v',k}\, r^{(n-1)}_{v,v'}.$$

Since $r^{(n)}_{u,u'}$ is computed by summing and multiplying non-negative terms, by induction, it is also non-negative. Furthermore, this value can be bounded from above as

$$\begin{aligned} r^{(n)}_{u,u'} &\leq \frac{(1-\gamma)^2}{d_u d_{u'}} \sum_{v,v'} r^{(n-1)}_{v,v'} \sum_{k \in L_V} \sum_{k' \in L_V} \pi_{v,k} \pi_{v',k'} \\ &= \frac{(1-\gamma)^2}{d_u d_{u'}} \sum_{v,v'} r^{(n-1)}_{v,v'} \left(\sum_k \pi_{v,k}\right) \left(\sum_{k'} \pi_{v',k'}\right) \\ &= \frac{(1-\gamma)^2}{d_u d_{u'}} \sum_{v,v'} r^{(n-1)}_{v,v'}. \end{aligned}$$

Let $r^{(n-1)}_{\max} = \max\limits_{v,v' \in V} r^{(n-1)}_{v,v'}$, we have

$$\begin{aligned} r^{(n)}_{u,u'} &= (1-\gamma)^2\, r^{(n-1)}_{\max} \\ &\leq (1-\gamma)^{2n}\, r^{(0)}_{\max} \qquad \text{(by induction)} \\ &= (1-\gamma)^{2n}\, \gamma^2. \end{aligned}$$

Finally, the probability of generating the same sequence of any length between u and u' is bounded by

$$\begin{aligned} \sigma_{u,u'} &= \sum_{n=1}^{\infty} r^{(n)}_{u,u'} \\ &\leq \gamma^2 \sum_{n=1}^{\infty} (1-\gamma)^{2n} \\ &= \frac{\gamma^2 (1-\gamma)^2}{1-(1-\gamma)^2}, \end{aligned}$$

where we have used the fact that the series is geometric and $(1-\gamma)^2 < 1$.

Proposition 2. *The kernel defined by (4) converges for any* $0 < \gamma \leq 1$.

Proof. This can be shown using the same approach as with Proposition 1.