Modeling Relationship Strength in Online Social Networks

Rongjing Xiang, Jennifer Neville
Computer Science Department
Purdue University
West Lafayette, IN 47907
[rxiang|neville]@cs.purdue.edu

Monica Rogati
LinkedIn
Mountain View, CA 94043
mrogati@linkedin.com

Abstract

Previous work analyzing social networks has mainly focused on binary friendship relations. However, in online social networks the low cost of link formation can lead to networks with heterogeneous relationship strengths (e.g., acquaintances and best friends mixed together). In this case, the binary friendship indicator provides only a coarse representation of relationship information. In this work, we develop an unsupervised model to estimate relationship strength from interaction activity (e.g., communication, tagging) and user similarity. More specifically, we formulate a link-based latent variable model, along with a coordinate ascent optimization procedure for the inference. We evaluate our approach on real-world data from Facebook, showing that the estimated link weights result in higher autocorrelation and lead to improved classification accuracy.

1 Introduction

Recent research on analyzing social networks has demonstrated that relational patterns of *homophily* [13] can be exploited to improve predictive models of both link structure and behavior. For example, researchers have found that network connectivity and attribute similarity can improve link prediction models [12, 17]. Also, researchers have found that relational ties can improve behavior prediction in tasks such as as fraud detection [14] and viral marketing [5]. However, much of this past work has focused on social networks with *binary* relational ties (e.g., friends or not). These binary indicators provide only a coarse indication of the nature of the relationship. Due to the low-cost of friendship identification in online social networks and the variance of link information in electronic communication networks, the resulting networks contain both strong and weak ties—with little or no information to differentiate between the two ends of the spectrum. Since pairs of individuals with *strong ties* (e.g., close friends) are likely to exhibit greater similarity than those with *weak ties* (e.g., acquaintances) [7], treating all relationships equally will increase the level of noise in the learned models and likely lead to degradation in performance.

Fortunately, online social networks (OSNs) often consist of more than just a record of social network ties. Typically online communities contain ancillary *interaction* information among the users that can be used to improve modeling. Indeed, almost every OSN provides infrastructure for the formation and maintenance of communities over time by facilitating communication and transfer of information. This interaction information records low-level transactions among related people and can be used to identify which linked members are close friends/colleagues, as opposed to acquaintances. Facebook users, for instance, each have a *Wall* page as part of their profile, where friends can post messages. While a particular user may have hundreds of friends, due to resource constraints it is likely that she communicates more frequently with stronger friends than acquaintances.

In this work, we propose a model to infer relationship strength based on profile similarity and interaction activity, with the goal of automatically distinguishing strong relationships from weak ones.

Recently, interaction data has been used to predict relationship strength [10, 6] but this work only considered two levels of relationship strength, namely weak and strong relationships. In addition, this past work focused on supervised learning methods, which requires human annotation of link strength (e.g., friendship rating [6] or top friend nomination [10]). We focus instead on developing a richer model that can represent the full spectrum of relationship strength, from weak to strong, and propose an *unsupervised* method to infer a continuous-valued relationship strength for links.

More specifically, we formulate a latent variable model to infer (hidden) relationship strengths and develop a coordinate ascent optimization procedure for inference. From the modeling perspective, a unique characteristic of our model is that it distinguishes interaction activity from users' profile data and it integrates these two types of information by considering the relationship strength to be the hidden effect of user background similarities, as well as the hidden cause of the interactions between users. This naturally results in a latent variable model which captures the causality of the underlying social process.

We outline the details of the model and the estimation algorithm in Section 2. In Section 3, we evaluate our approach on real-world data from Facebook by showing that autocorrelation in the estimated relationship-strength graph is higher than any alternative graph formed from various aspects of the raw data. Furthermore, we demonstrate the utility of the inferred relationship strengths in a number of collective classification tasks.

2 Latent variable model

One key assumption underlying our model is the theory of *homophily* from sociology [13]. The homophily principle postulates that people tend to form ties with other people who have similar characteristics (i.e., the tendency of like to associate with like). Moreover, it is likely that the stronger the tie, the higher the similarity [7]. Previous studies have shown that homophily is ubiquitous in social networks [13]. In online social networks therefore, we can model the relationship strength as a *hidden effect* of nodal profile similarities. Such profile attributes include, for instance, the schools and companies the users attended, the online groups that they joined, the geographic locations that they belong to, etc.

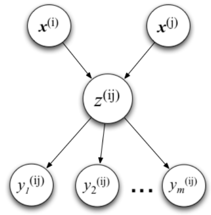
Furthermore, we assume that the relationship strength directly impacts the nature and frequency of online interactions between a pair of users. Since each user has a finite amount of resources (e.g., time) to use in the formation and maintenance of relationships, it is likely that they direct these resources towards the relationships that they deem more important [4]. Such interactions could be, for example, profile viewing activities between the pair of users, connection establishment, picture tagging, etc. The stronger the relationship, the higher likelihood that a certain type of interaction will take place between the pair of users. In this way, we model the relationship strength as the *hidden cause* of user interactions.

Formally, let $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(j)}$ be the attribute vectors of two individuals i and j, and let $y_t^{(ij)}, t = 1, 2, \ldots, m$ be the m different interactions considered between i and j. Then we define $z^{(ij)}$ to be the latent relationship strength between i and j and model the influence of $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(j)}$ on $z^{(ij)}$, as well as the influence of $z^{(ij)}$ on $y_t^{(ij)}, t = 1, 2, \ldots, m$. We illustrate the general model using the directed graphical model representation in Figure 1(a). The full model can be viewed as a hybrid of discriminative and generative models—the upper part is discriminative ($p(Z|X)$), while the lower part ($p(Y|Z)$) is generative. Our model represents the likely causal relationships among these variables by modeling the conditional dependencies, so that the joint distribution decomposes as follows:

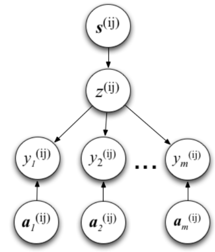
$$P(z^{(ij)}, \mathbf{y}^{(ij)}|\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = P(z^{(ij)}|\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \prod_{t=1}^{m} P(y_t^{(ij)}|z^{(ij)}) \tag{1}$$

Although the relationship-strength variable z summarizes the similarities and interactions between a pair of people, its value is not directly observed in the data (and it would be difficult, if not impossible, to collect from online users). As such, it makes sense to treat z as a latent (i.e., hidden) variable in the model, which we will estimate for each pair of people (along with the values of model parameters) so as to maximize the overall observed data likelihood.

In general, our model can be applied to infer either directed or undirected relationship strengths, depending on the way how we specify the profile similarity and the interactions for each pair. In this



(a)



(b)

Figure 1: Graphical model representation of (a) the general relationship strength model, and (b) the specific instantiation described in Sec.2.1.

work, we infer directed relationship strengths, i.e., the estimated $z^{(ij)}$ could possibly be different than $z^{(ij)}$, since we consider directed interactions in the data (e.g., while i has posted on j's wall, j might not have posted on i's wall).

2.1 Model specification

The general latent variable model of relationship strength can be instantiated in different ways, depending on domain-specific availability and interpretation of attributes and interactions. In this work, we adopt the widely-used Gaussian distribution to model the conditional probability of the relationship strength given the attribute similarities. Let $s_k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})(k = 1, 2, \ldots, n)$ denote a set of similarity measures taken on the pair. Then the dependency between z^{ij} and $\mathbf{x}^{(i)}, \mathbf{x}^{(j)}$ is:

$$P(z^{(ij)}|\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \mathcal{N}(\mathbf{w}^T \mathbf{s}(\mathbf{x}^{(j)}, \mathbf{x}^{(j)}), v) \tag{2}$$

where $\mathbf{s}$ is a similarity vector calculated based on $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(j)}$; $\mathbf{w}$ is an n-dimensional weight vector to be estimated; v is the variance in Gaussian model, which is configured to be 0.5 in our experiments. The probability distribution of each $y_t^{(ij)}$ given $z^{(ij)}$ is conditionally independent. For this work, due to data sparsity, we model all interactions as binary variables, regardless of the frequency of interaction. For example, $y_t^{(ij)}$ may denote whether a user i has posted on j's wall.

Furthermore, to increase the accuracy of the model, we introduce auxiliary variables $a_{t1}^{(ij)}, a_{t2}^{(ij)}, \ldots, a_{tl_t}^{(ij)}$ for each interaction t, as shown in Figure 1(b). Such variables capture auxiliary causes of the interactions which are independent of the relationship strength. For example, the total number of pictures that a user has tagged represents their intrinsic tendency to tag pictures. We use logistic function can be used to model the conditional probability of $y_t^{(ij)}$ given $z^{(ij)}$ and $\mathbf{a}_t^{(ij)}$.

$$P(y_t^{(ij)} = 1|\mathbf{u}_t^{(ij)}) = \frac{1}{1 + e^{-(\boldsymbol{\theta}_t^T \mathbf{u}_t^{(ij)} + b)}} \tag{3}$$

where we have defined $\mathbf{u}_t^{(ij)} = \begin{bmatrix} \mathbf{a}_t^{(ij)} \\ z^{(ij)} \end{bmatrix}$ for a more compact notation, and $\boldsymbol{\theta}_t = [\theta_{t,1}, \theta_{t,2}, \ldots, \theta_{t,l_t+1}]^T$ is the set of parameters to be estimated.

Note that in general, we could apply other appropriate generalized linear models for interaction variables without adding difficulty to the inference since the objective function will remain concave. For example, poisson regression could be used if the interactions are represented as count data. Finally, to avoid over-fitting, we put L2 regularizers on the parameters $\mathbf{w}$ and θ, which can be regarded as Gaussian priors.

The data are represented as N user pair samples, denoted by $\mathcal{D} = \{(i_1, j_1), (i_2, j_2), \ldots, (i_N, j_N)\}$. During training, The variables $\mathbf{x}^{(ij)}$, $\mathbf{y}^{(ij)}$ and $\mathbf{a}_t^{(ij)}$, $((ij) \in \mathcal{D}, t = 1, 2, \ldots, m)$ are all visible. Since the attribute similarities are pre-calculated based on the $\mathbf{x}'s$, to simplify the notation, we define $s^{(ij)} = s(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$. Given all the observed variables, based on Eq. (1), the joint probability is as follows:

$$P(\mathcal{D}|\mathbf{w}, \boldsymbol{\theta})P(\mathbf{w}, \boldsymbol{\theta}) \tag{4}$$

$$= \left(\prod_{(i,j)\in\mathcal{D}} P(z^{(ij)} P(z^{(ij)}|x^{(i)}, x^{(j)}, \mathbf{w}) \prod_{t=1}^{m} P(y_t^{(ij)}|z^{(ij)}, \boldsymbol{\theta}_t) \right) P(\mathbf{w})P(\boldsymbol{\theta}_t)$$

$$\propto \left(\prod_{(i,j)\in\mathcal{D}} \left(e^{-\frac{1}{2v}\left(\mathbf{w}^T\mathbf{s}^{(ij)}-z^{(ij)}\right)^2} \prod_{t=1}^{m} \frac{e^{-(\boldsymbol{\theta}_t^T\mathbf{u}^{(ij)}+b)(1-y_t^{(ij)})}}{1+e^{-(\boldsymbol{\theta}_t^T\mathbf{u}_t^{(ij)}+b)}} \right) \right) e^{-\frac{\lambda_w}{2}\mathbf{w}^T\mathbf{w}} \prod_{t=1}^{m} e^{-\frac{\lambda_\theta}{2}\boldsymbol{\theta}_t^T\boldsymbol{\theta}_t}$$

2.2 Inference

In general, estimation of a latent variable model can be done in two different ways. First, we can infer the distribution of the latent variable z, and find point estimates of the parameters $\hat{\mathbf{w}}, \hat{\boldsymbol{\theta}}$ so as to maximize the joint likelihood $P(\mathbf{y}, \hat{\mathbf{w}}, \hat{\boldsymbol{\theta}}|\mathbf{x})$ (i.e., the latent variable $\mathbf{z}$ is integrated out). This type of approach usually involves an iterative expectation maximization (EM) algorithm. Second, we can treat the latent variable as a parameter—namely, find point estimates $\hat{\mathbf{w}}, \hat{\boldsymbol{\theta}}, \hat{\mathbf{z}}$ that maximize the likelihood $P(\mathbf{y}, \hat{\mathbf{z}}, \hat{\mathbf{w}}, \hat{\boldsymbol{\theta}}|\mathbf{x})$. In this work, we will use the latter approach since integration over the latent variable z involved in the E-step of an EM algorithm would be intractable. We leave the investigation of an approximate EM algorithm, to estimate the distribution of the latent variables z, as future work.

Taking the logarithm of Eq. (4), we get the data log-likelihood:

$$\begin{aligned} &\mathcal{L}(z^{(\{(i,j)\in\mathcal{D}\})}, \mathbf{w}, \boldsymbol{\theta}_t) \qquad (5) \\ =& \sum_{(ij)\in\mathcal{D}} \left(-\frac{1}{2v}\left(\mathbf{w}^T\mathbf{s}^{(ij)} - z^{(ij)}\right)^2 + \sum_{t=1}^{m}\left(-(1-y_t^{(ij)})(\boldsymbol{\theta}_t^T\mathbf{u}_t^{(ij)}+b) - \log\left(1+e^{-\left(\boldsymbol{\theta}_t^T\mathbf{u}_t^{(ij)}+b\right)}\right)\right)\right) \\ &- \frac{\lambda_\mathbf{w}}{2}\mathbf{w}^T\mathbf{w} - \sum_{t=1}^{m}\frac{\lambda_\theta}{2}\boldsymbol{\theta}_t^T\boldsymbol{\theta}_t + C \end{aligned}$$

Note that in Eq. (5), both the quadratic terms and the logarithm of logistic function are concave. Since the sum of concave functions is concave, the function $\mathcal{L}$ is concave. Therefore, a gradient-based method will allow us to we optimize over the parameters $\mathbf{w}, \boldsymbol{\theta}_t$, $(t = 1, 2, \ldots, m)$ and the latent variables $z^{(ij)}, (ij) \in \mathcal{D}$ to find the maximum of $\mathcal{L}$. Below, we derive a coordinate ascent method for the optimization. An overview of the optimization procedure is given in Table 1.

The coordinate-wise gradients are:

$$\frac{\partial\mathcal{L}}{\partial z^{(ij)}} = \frac{1}{v}(\mathbf{w}^T\mathbf{x}^{(ij)} - z^{(ij)}) + \sum_{t=1}^{m}\left(y_t^{(ij)} - \frac{1}{1+e^{-(\boldsymbol{\theta}_t^T\mathbf{u}_t^{(ij)}+b)}}\right)\theta_{t,l_t+1} \qquad (6)$$

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\theta}_t} = \sum_{(ij)\in\mathcal{D}}\left(y_t^{(ij)} - \frac{1}{1+e^{-(\boldsymbol{\theta}_t^T\mathbf{u}_t^{(ij)}+b)}}\right)\mathbf{u}_t^{(ij)} - \lambda_{\boldsymbol{\theta}}\boldsymbol{\theta}_t \qquad (7)$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}} = \frac{1}{v}\sum_{(ij)\in\mathcal{D}}(z^{(ij)} - \mathbf{w}^T\mathbf{s}^{(ij)})\mathbf{s}^{(ij)} - \lambda_w\mathbf{w} \qquad (8)$$

A coordinate ascent optimization scheme will update $\mathbf{w}, z^{(ij)}, \boldsymbol{\theta}_t$ iteratively until convergence. For $z^{(ij)}$ and $\boldsymbol{\theta}_t$, we use Newton-Raphson update in each iteration, where the 2nd order derivatives are given by:

$$\frac{\partial^2\mathcal{L}}{\partial\left(z^{(ij)}\right)^2} = -\frac{1}{v} - \sum_{t=1}^{m}\frac{\theta_{t,l_i+1}^2 e^{-(\boldsymbol{\theta}_t^T\mathbf{u}_t^{(ij)}+b)}}{\left(1+e^{-(\boldsymbol{\theta}_t^T\mathbf{u}_t^{(ij)}+b)}\right)^2} \qquad (9)$$

$$\frac{\partial^2\mathcal{L}}{\partial\boldsymbol{\theta}_t\partial\boldsymbol{\theta}_t^T} = -\sum_{(ij)\in\mathcal{D}}\frac{e^{-(\boldsymbol{\theta}_t^T\mathbf{u}_t^{(ij)}+b)}}{\left(1+e^{-(\boldsymbol{\theta}_t^T\mathbf{u}_t^{(ij)}+b)}\right)^2}\mathbf{u}_t^{(ij)}\mathbf{u}_t^{(ij)^T} - \lambda_\theta\mathbf{I} \qquad (10)$$

For $\mathbf{w}$, the root of (8) can be found analytically as in usual ridge regression:

$$\hat{\mathbf{w}} = \left(\lambda_w\mathbf{I} + \mathbf{S}^T\mathbf{S}\right)^{-1}\mathbf{S}^T\mathbf{z} \qquad (11)$$

where $\mathbf{S} = \left[\mathbf{s}^{(i_1j_1)}\ \mathbf{s}^{(i_2j_2)}\ \ldots\ \mathbf{s}^{(i_Nj_N)}\right]^T$, and $\mathbf{z} = \left[z^{(i_1j_1)}\ z^{(i_2j_2)}\ \ldots\ z^{(i_Nj_N)}\right]^T$.

While not converged: 1. For each Newton-Raphson step: For $t = 1, \ldots, m$: **update** $\boldsymbol{\theta}_t$ based on Eq. (7) and Eq. (10). 2. For each Newton-Raphson step: For $(i, j) \in \mathcal{D}$: **update** $\mathbf{z}^{(ij)}$ based on Eq. (6) and Eq. (9). 3. Update $\mathbf{w}$ based on Eq. (11)

Table 1: The learning algorithm

During testing, for a new pair of users (i, j), the learned model can be applied in two ways. First, if both user attributes $\mathbf{x}^{(i)}, \mathbf{x}^{(j)}$ and their interactions $y_1^{(ij)}, \ldots, y_t^{(ij)}$ are known, we can estimate the relationship strength $z^{(ij)}$ in the same way as step 2 in the learning algorithm. Second, when the interaction data are unobserved, we just apply Eq. (2) to infer $z^{(ij)}$. Since the interaction data are usually sparse, temporal, and difficult to obtain, the second scenario is more common in real online social networks. This in fact demonstrates a strength of our hybrid model: the lower part of the model is generative so that the overall model will not suffer much from missing interaction data during testing. Once the model is learned, for new instances the latent variables can be inferred using only the upper level of variables in the model. In addition, the generative lower part also facilitates application of the learned model for predicting future interactions (e.g., predicting new connections). On the other hand, fully discriminative models which treat user background information and interactions equally, will not be easy to apply in these situations.

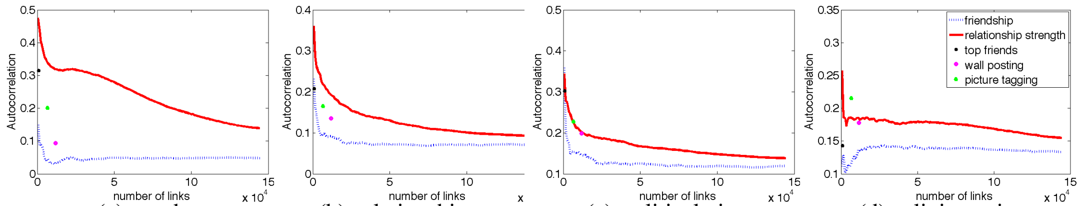
3 Experiments

We evaluate our proposed model on data from the public Purdue Facebook network (www.facebook.com). For these experiments, we randomly selected five Purdue users as seed nodes and considered all nodes within three hops of the seed nodes, in the Purdue friendship graph, which resulted in a total sample of 4500 nodes. From this sample, we constructed a training set of all the directly linked users, which amounts to 144,712 pairs.

For each pair of users (i, j), we computed three features to capture the similarity among their profiles and their connections. These features are: the logarithms of the normalized counts of common networks, common groups, and common friends that i and j share. Note here that for evaluation purposes, we do not consider similarity features computed from user profile attributes (namely, gender, relationship status, political and religious views) and use only information about network and group membership, and connection topology in our similarity features. We do this for two reasons. First, we will later use profile similarity to evaluate the quality of the estimated link strengths, so for accurate evaluation we cannot use these features during learning. Second, in the publicly available Facebook data, the majority of users either do not list these profile attribute or they do not make the information public (e.g., only 44% have gender and 27% have political views listed in their public profiles). In addition to similarity features, we also considered two types of user interactions in the model: whether i has posted on j's wall, and whether i has tagged j in photos. Furthermore, we also include in the model the total number of people that i has wall-posted to or picture-tagged as the auxiliary variable for the corresponding interaction variable.

For evaluation, we compare the weighted graph formed by the estimated relationship strengths to the following four graphs formed from the observed data. (1) **Friendship graph:** The graph consists of all friendship links between users. This network can be viewed as a graph of both "strong" and "weak" relationships. (2) **Top-friend graph:** The graph consists of all *top-friend* nominations. Facebook has a "Top Friends" application which allows users to nominate a small portion of their friends as *best* friends. Such top-friend links can be regarded as "strong" relationships but the resulting network is quite sparse [10]. (3) **Wall graph:** The graph consists of edges corresponding to wall posting activities. Every link corresponds to a pair of users (i, j) such that i has posted on j's wall. This network can be viewed as an interaction network. (4) **Picture graph:** The graph consists of edges corresponding to picture-tagging activities. Every link corresponds to a pair of users (i, j) such that i has tagged j in his or her uploaded pictures. This network can also be viewed as an interaction network.

Autocorrelation improvement. In relational data, autocorrelation is the statistical dependency of the same attribute on related instances [9]. To measure the autocorrelation of a categorical attribute



(a) gender (b) relationship status (c) political views (d) religious views

Figure 2: Autocorrelation on various graphs, as link density is varied. For the sparse networks (i.e., top-friend, wall, picture), we plot a single point for the observed autocorrelation. For the relationship-strength graph, we vary the number of links in the network by thresholding the link strength and plot the associated autocorrelation as the number of links is increased. For comparison, on the friendship graph we randomly drop links to assess the autocorrelation on networks with the same density. Note that the maximum value on the x-axis reflects the autocorrelation with *all* the friendship-links/relationship-strengths in the network.

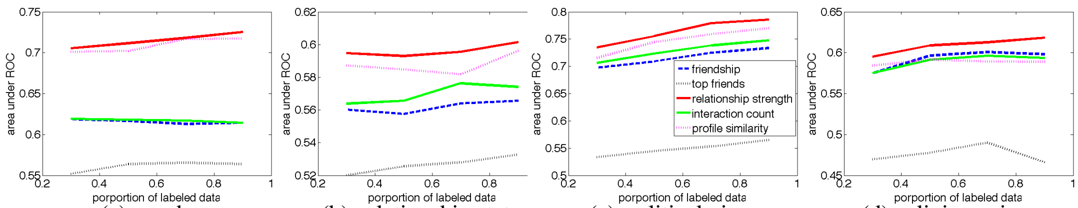
A with K values in a graph, we first construct a contingency table which consists of K rows and K columns where each row (or column) corresponds to a value of A, and then the autocorrelation is calculated based on the chi-square statistic: $\chi^2 = \sum_{a \in K} \sum_{b \in K} \frac{O_{ab} - E_{ab}}{E_{ab}}$, where E_{ab} is the expected occurrence of the attribute value pair $(i.a, j.b)$, and O_{ab} is the observed occurrence of the attribute value pair $(i.a, j.b)$ across pairs of linked nodes (i, j) in the graph. We scale the chi-square statistic to the range $[-1, 1]$ by using the corrected contingency coefficient: $CC = \sqrt{\frac{K\chi^2}{(K-1)(N+\chi^2)}}$, where N is the number of linked pairs in the data. For the weighted graph, we scale the counts in each cell by the link strength of the corresponding link (N is scaled as well).

In Figure 2 we graph the autocorrelation of the withheld profile attributes (i.e., gender, relationship status, political and religious views) on the five networks. The graphs show that, with the exception of religious views in the picture network, the relationship-strength network has autocorrelation greater than, or equal to, all other networks (for the same number of links). In particular, the relationship-strength network has significantly higher autocorrelation than the friendship graph in all cases—even though these four profile attributes were not used to learn the model. This demonstrates the utility of the model to identify relationships that reflect the natural similarity among people. Also, we note that as the number of links increases, for both the friendship and relationship-strength networks, the autocorrelation decreases. This indicates a tradeoff between link density and autocorrelation—as we restrict the number of relationships it is likely that similarity among users increase, however increased sparsity will hamper our ability to use the autocorrelation to improve predictive models. We will revisit this issue in the classification discussion below.

Classification improvement. We further explore the utility of the weighted graph formed from relationship strength by applying it in a number of collective classification tasks. For each of the four profile attributes, we considered a binary classification task based on its most frequent value. We apply a widely-used semi-supervised classification algorithm—the Gaussian random field (GRF) model [18], which assumes autocorrelation is present in the graph and propagates information from the labeled portion of the graph to infer the values for unlabeled nodes. Since the GRF is designed for application on undirected graphs, we modify each link $w(i, j)$ in the four directed graphs (relationship strength, top-friend, wall-posting and picture-tagging graphs) to be $\max\{w(i, j), w(j, i)\}$. We vary the proportion of labeled nodes in the graph from 30% to 90% and measure the resulting classification rankings using area under the ROC curve.

The classification results are shown in Figure 3. The performance curves for the wall graph and the picture graph lie well below the interaction-count graph for all classification tasks so we omit them in the plot for clarity. Note the poor performance of the top-friend graph—this occurs despite the fact that the top-friend network had the highest autocorrelation of the observed graphs for all but religious views, which indicates that high autocorrelation is only helpful for classification if the network has sufficient density to exploit the correlation among neighbors. This illustrates one strength of our proposed approach, since the model can maintain the density of the full friendship graph while significantly increasing the autocorrelation levels.

Indeed, the relationship-strength graph results in the highest classification performance for all tasks, suggesting that our approach to summarizing the rich profile and interaction information in online social networks results in a single meaningful *relationship* graph which can improve subse-



(a) gender (b) relationship status (c) political views (d) religious views

Figure 3: Classification using the relationship-strength graph is compared with the four observed Facebook graphs (friendship, top-friend, wall, picture), as well as two additional graphs: (1) a profile-similarity graph, which weights each link by $\sum_{k=1}^{N} s_k^{(ij)}$, and (2) an interaction-count graph, which sums the links in the wall, top-friend, and picture graphs. The profile-similarity and interaction-count graphs are natural heuristics to illustrate the utility of the upper portion (profile similarity) and lower portion (interaction occurrence) of our proposed model in isolation. All results are averaged over five runs with different random selections of labeled instances.

quent knowledge discovery and prediction tasks. We note that neither the profile-similarity nor the interaction-count graphs perform well across all the tasks. This illustrates another strength of our proposed approach, since we combine both these sources of relationship information together in a single representation of overall relationship strength.

4 Related Work

The recent growth and popularity of online social networks (OSNs) such as Facebook, MySpace, and LinkedIn has lead to a surge in research focused on modeling networks and their properties. Much of this work has focused on the analysis of network structure and growth patterns (see e.g., [2, 3, 16]). However, nearly all these methods focus on descriptive analysis and generative models of link structure, based on the observed structure in the network—they do not attempt to model the latent properties of the networks.

Another direction of related research has focused on link prediction—which is a formulization of the problem of predicting future links in a social network, given a snapshot of the network at the current time step. This is the area of research that is most relevant to our work in this paper. Link prediction methods can be generally grouped into two approaches: those that use *topological* features to capture the link structure of the network (e.g., [12, 11]) and those that use attribute similarity features in addition to topological features (e.g., [17, 8]).

We differ from past work on link prediction in that we focus on modeling link strength rather than link existence. We also aim to exploit interaction information among nodes in order to improve model accuracy. O'Madadhain et al. [15] also model interaction events, but they formulate a temporal link prediction task which tries to predict the occurrence of events (e.g., co-authorship) in future time intervals. Adamic and Adar [1] also investigate the use of ancillary network information but with the goal of predicting social ties, instead of tie strength. More recently, interaction data has been utilized in predicting relationship strength [10, 6], but this work has considered the binary prediction task of distinguishing strong ties from weak ties. Our work instead uses a richer representation that can span the full spectrum from weak to strong ties. Moreover, this previous work has focused on supervised methods, which usually involve efforts on human annotation (e.g., friendship rating [6] or top friend nomination [10]). Our method takes an unsupervised approach instead, inferring a continuous-valued measure of relationship strength for online social networks.

5 Conclusions and Future Work

In this paper, we proposed a latent variable model for the task of relationship strength estimation in online social networks. The model attempts to represent the intrinsic causality of social interactions via statistical dependencies. Our experiments show that the weighted graph formed by the estimated relationship strengths gives rise to higher autocorrelation and better classification performance than the graphs formed from various aspects of the raw data.

There are a number of modifications to our proposed model that can be explored in future work. Since the current inference scheme uses point estimation for the latent variable, we plan to develop alternative inference procedures which maximize the observed data likelihood by integrating over all possible values of the latent variable. This will involve the use of a mathematically convenient approximate distribution on the latent variable. Furthermore, we will consider alternative ways to specify the model—for instance, we could apply kernels in defining profile similarities and learn the functions automatically. Also, nonlinear classification or regression could be used instead of the current choice of a generalized linear model for the interaction dependencies. In both cases, however, the difficulty of inference will increase.

There are also several directions for future work that we will investigate as natural extensions of this work. First, the current model considers the relationship strength on each edge independently. Although inference would be more complex, it may improve the accuracy of the model if we consider the dependencies between adjacent edges. Second, it would be interesting to investigate the evolutionary aspect of relationship strengths over time. Finally, we will consider other interesting applications which use relationship strength to understand human behavior.

Acknowledgments

This research is supported by DARPA and NSF under contract numbers NBCH1080005, SES-0823313, and IIS- 0916686. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation hereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements either expressed or implied, of DARPA, NSF, or the U.S. Government.

References

[1] A. Adamic and E. Adar. Friends and neighbors on the web. *Social Networks*, 25(2):211–230, 2003.

[2] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan. Group formation in large social networks: membership, growth, and evolution. In *KDD '06*, 2006.

[3] D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri. Feedback effects between similarity and social influence in online communities. In *KDD '08*, 2008.

[4] K. Dindia and D. Canary. Definitions and Theoretical Perspectives on Maintaining Relationships. *Journal of Social and Personal Relationships*, 10(2):163–173, 1993.

[5] P. Domingos and M. Richardson. Mining the network value of customers. In *KDD '01*, 2001.

[6] E. Gilbert and K. Karahalios. Predicting tie strength with social media. In *CHI '09*, 2009.

[7] M. Granovetter. The strength of weak ties: A network theory revisited. *Sociological Theory*, 1:201–233, 1983.

[8] M. Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *Proceedings of the Workshop on Link Discovery: Issues, Approaches and Applications*, 2005.

[9] D. Jensen and J. Neville. Linkage and autocorrelation cause feature selection bias in relational learning. In *ICML '02*, 2002.

[10] I. Kahanda and J. Neville. Using transactional information to predict link strength in online social networks. In *ICWSM '09*, 2009.

[11] H. Kashima and N. Abe. A parameterized probabilistic model of network evolution for supervised link prediction. In *ICDM '06*, 2006.

[12] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM '03*, 2003.

[13] M. McPherson, L. Smith-Lovin, and J. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001.

[14] J. Neville, O. Simsek, D. Jensen, J. Komoroske, K. Palmer, and H. Goldberg. Using relational knowledge discovery to prevent securities fraud. In *KDD '05*, 2005.

[15] J. O'Madadhain, J. Hutchins, and P. Smyth. Prediction and ranking algorithms for event-based network data. *SIGKDD Explorations*, 7(2):23–30, 2005.

[16] P. Singla and M. Richardson. Yes, there is a correlation: - from social networks to personal behavior on the web. In *WWW '08*, 2008.

[17] B. Taskar, M. F. Wong, P. Abbeel, and D. Koller. Link prediction in relational data. In *NIPS '03*, 2003.

[18] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML '03*, 2003.