Exploiting Time-Varying Relationships in Statistical Relational Models

Umang Sharan
Department of Computer Science
Purdue University
West Lafayette, IN
usharan@cs.purdue.edu

Jennifer Neville
Departments of Computer Science and Statistics
Purdue University
West Lafayette, IN
neville@cs.purdue.edu

ABSTRACT

In a growing number of relational domains, the data record temporal sequences of interactions among entities. For example, in citation domains authors publish scientific papers together each year and in telephone fraud detection domains people make calls to each other each day. The temporal dynamics of these interactions contain information that can improve predictive models (e.g., people publishing together frequently are likely to be publishing on the same topic) but to date there has been little effort to incorporate time-varying dependencies into relational models. Past work in relational learning has focused primarily on static "snapshots" of relational data. In this paper, we present an initial approach to modeling dynamic relational data graphs in predictive models of attributes. More specifically, we use a two-step process that first summarizes the dynamic graph with a weighted static graph and then incorporates the link weights in a relational Bayes classifier. We evaluate our approach on the Cora dataset (where co-author and citation links vary over time) showing that our approach results in significant performance gains over a baseline snapshot approach that ignores the temporal component of the data.

1. INTRODUCTION

Recent research has demonstrated the utility of modeling relational information in domains such as fraud detection [14], citation analysis [11], and marketing [5]. However, this work has focused primarily on static snapshots[1] of relational datasets, even though most relational domains have temporal dynamics that are important to model. For example, in fraud detection it is informative to know who is working with whom, but it may also be helpful to model temporal patterns of communications between suspects. Similarly, in marketing domains it is informative to know which customers are affiliated, but it may also be helpful to model how those associations change over time. To date, there are few available data mining tools that can simultaneously exploit both the temporal and relational aspects of these domains.

Recent work in the area of statistical relational learning has focused on transforming temporal data into aggregated features [14] or on modeling the temporal dynamics of time-varying attributes [18]. There have also been efforts to model temporally-varying links to improve automatic discovery of relational communities or groups [1, 8] but this work has not attempted to use these communities in a classification context. In this work, we incorporate time-varying links into statistical relational models in order to improve attribute prediction in dynamic domains.

One motivation for modeling time-varying links is the recent performance improvement that has been realized through exploiting homophily[2] in relational domains. The presence of homophily offers a unique opportunity to improve model performance because inferences about one object can be used to improve inferences about related objects. For example, if we know one person is involved in fraudulent activity, then his associates have increased likelihood of being engaged in misconduct as well [3]. Indeed, recent work in relational modeling has shown that *collective inference* over an entire dataset can result in more accurate predictions than conditional inference for each instance independently (e.g., [2]) and that the gains over conditional models increase as homophily increases [9].

The accuracy of collective inference techniques, however, is contingent on the presence of links in the data that confer homophily. In real-world datasets that evolve over time, it is quite likely that recent links confer more homophily than links in the distant past. Also, a protracted series of events between two entities can indicate a stronger underlying relationship and thus one that is more likely to confer homophily. We conjecture that temporal sequences of interactions among entities provide information about the nature of relationships between entities and that these interactions can be accurately and efficiently mined to identify higher-level relationships that confer more homophily than individual links.

As a first step towards mining these temporal patterns, we outline and investigate a new approach to incorporating time-varying link information into statistical relational models. We use a two-step process that first transforms a

[1]A snapshot at time t consists of all the objects, links, and attributes that have occurred up to and including time t.

[2]The tendency of like to associate with like [12].

Joint 9th WEBKDD and 1st SNA-KDD Workshop '07, August 12, 2007, San Jose, California, USA . Copyright 2007 ACM 1-59593-444-8...$5.00.

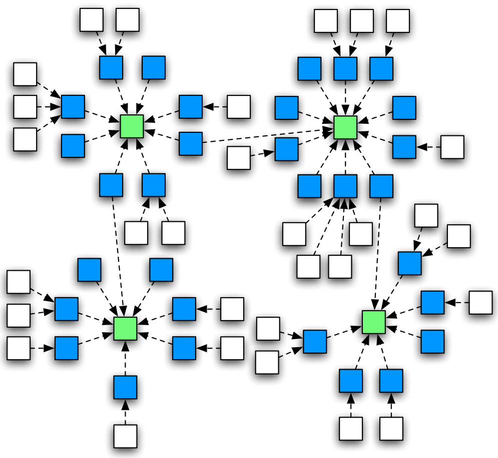

Figure 1: Example relational data fragment.

dynamic relational graph into a static weighted summary graph, based on recent work modeling *communities of interest* [3, 8]. Next the link weights are incorporated into a relational classifier to moderate the influence of attributes throughout the relational data graph.

We evaluate our algorithm on the Cora dataset—a database of computer science research papers extracted automatically from the web [10]. We model the temporally-varying citation and coauthor relationships with our approach and compare to a baseline snapshot model that ignores the temporal component of the data. We demonstrate that modeling the temporal component of the relational structure significantly improves classification performance.

2. RELATIONAL DATA

We will consider relational data represented as an attributed multi-graph. More specifically, a directed, attributed graph $G = (V, E)$, with V nodes representing objects and E edges representing relations[3], with one or more connected components. For example, consider the data graph in Figure 1. The nodes V represent objects in the data (e.g., people, organizations, events) and the edges E represent relations among the objects (e.g., works-for, located-at).

Each node $v_i \in V$ and edge $e_j \in E$ are associated with a type $G(v_i) = g_{v_i}$, $G(e_j) = g_{e_j}$. This is depicted by node color in Figure 1. For example, blue nodes represent *people* and green nodes represent *organizations*. Each object or link type $g \in G$ has a number of associated attributes $\mathbf{X^g} = (X_1^g, ..., X_m^g)$ (e.g., age, gender).

When relational data has a temporal component, there are three aspects of the data that may change over time. First, there may be attributes whose values that vary over time $\mathbf{X_i} = (X_{i1}, X_{i2}, ..., X_{it})$. Second, there may be relationships that vary over time. This results in a different data graph $G_t = (V, E_t)$ for each time step t, where the nodes remain constant but the edge set may vary (i.e., $E_{ti} \neq E_{tj}$ for some i,j). Third, there may be objects that appear and disappear over time. This is also represented as a set of data graphs $G'_t = (V_t, E_t)$, but in this case both the objects and links sets may vary.

[3]Note that there may be more than one edge between a pair of nodes.

Initial efforts to incorporate time into statistical relational models have focused on either the first or the third case. For domains, where attributes vary over time, Sanghai et. al [18] have extended probabilistic relational models (PRMs) [6] in a manner similar to dynamic Bayes networks [7], rolling out a separate PRM for each time step and modeling the dependencies of attribute values in one time step to the next. There is some recent work on modeling domains where the number of objects is uncertain and can change over time [13]. However, to our knowledge, there are no statistical relational models for domains where the links change over time.

In this paper we focus on the case where the links vary over time and outline a model for this type of dynamic domain. For example, consider the case where we have a set of people coauthoring scientific papers. In each year there will be different sets of people coauthoring different papers. This is represented in Figure 2. The nodes (authors) remain constant but the edges change in each time step, representing the coauthor events that have occurred in each year.

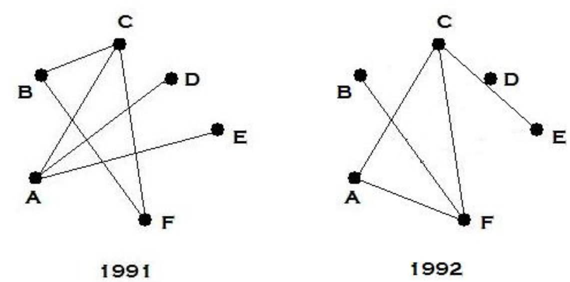

Figure 2: Example data graph changing over time.

3. ALGORITHM

Our approach to classification in domains where the links vary over time uses a two-step process that we describe in detail below. First, we summarize the set of temporally varying relational graphs into a static weighted summary graph. Second, the link weights are incorporated into a relational classifier to moderate the influence of attributes by the strength of the associated relationship over time. In this paper, we have chosen to use a relational Bayes classifier (RBC) [16] for its simplicity and to avoid any confounding effects due to feature selection. However, other relational classifiers may also be used (e.g., [17, 6, 15]), provided that they can handle weighted instances during learning and inference.

3.1 Graph summarization

A simple and efficient way to model temporally-varying relational networks is to transform a temporal sequence of data graphs $\mathbf{G_t} = (G_1, G_2, ..., G_t)$ into a static graph G^S by summarizing the sequence of links between any pair of nodes by a single edge with a weight. We employ this approach with an exponential weighting scheme based on the algorithm of

Cortes et al. [3]. We define the summary graph at time t to be the weighted sum of the graph at time t and the summary graph at time $t-1$, with a parameter θ to specify the influence of the current time step:

$$G_t^S = \begin{cases} (1-\theta)G_{t-1}^S + \theta G_t & \text{if } t > 1 \\ \theta G_t & \text{if } t = 1 \end{cases}$$

The above equation recursively defines the summary graph G_t^S at time t as the weighted average of the summary graph G_{t-1}^S at time $(t-1)$ and the data graph G_t at time t. More specifically, we use the weighted sum of the edge sets:

$$E_t^S = \{e_{ij} : e_{ij} \in E_{t-1}^S \vee e_{ij} \in E_t\}$$

$$and\ \ w_{e_{ij}^t} = (1-\theta) w_{e_{ij}^{t-1}} \cdot I_{E_{t-1}^S} + \theta \cdot I_{E_t}$$

where $I_E = |e_{ij}|$ if $e_{ij} \in E$ and 0 otherwise. Note that this recursive definition of the summary graph incorporates all links from time 1 to time t.

An example summary graph between for the data in Figure 2 is shown in Figure 3. Dotted edges denote diminishing weights, solid edges imply reinforced weights, while red edges denote fresh edges.

If we look at Figure 2, it is more likely that A and C are close collaborators (in 1992) than A and E or A and D since A published with C in both 1991 and 1992 but only published with E and D in 1991. Similarly, it is more likely that C and F have a stronger collaboration than C and E since C and E only started publishing together in 1992 while C and F have been publishing together since 1991. These are the notions that we are trying to capture in the exponential weighting scheme. Such relationship patterns are quite common in social network data— recent relationships tend to be stronger than old and forgotten ones, while old and perennial relationships tend to be the healthiest. We will model the influences of citations and coauthors in a similar manner for our predictive task. In other words, we postulate that a paper's topic is more likely to be influenced by a reference's topic that has been published recently rather than one that was published farther in the past. In addition, we postulate that an author is more likely to be working on a topic similar to those of his recent coauthors than those of his past coauthors.

Note that our approach is general enough to handle cases when more than one link exists between a pair of nodes at a given time step. However, it does assume that if there are multiple links that they are all of the same type and thus can be summarized into a single relationship. For example, if two authors A and B publish two papers and administer a grant together in year t, $I_E = 3$ and there will be one summary edge between A and B even though the coauthor links and the co-PI links do not necessarily indicate the same type of relationship between A and B. This case can be easily dealt with by partitioning the graph into a separate graph for each link type g before summarization. However, for the experiments below, we focus our analysis on a dataset where each pair nodes can only be linked by edges of a single type (i.e., paper-to-paper citation links, and person-to-person coauthor links).

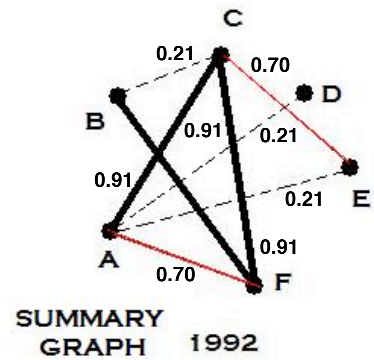

Figure 3: Summary graph at time $t = 1992$, $\theta = 0.7$ for the data in Figure 2.

3.2 Relational Bayes classifier

For learning and inference, we extend a relational Bayes classifier (RBC) [16] to incorporate the link weights from the summary graph into the probability calculations.

RBCs extend naive Bayes classifiers to a relational setting by treating heterogeneous relational subgraphs as a homogenous set of attribute multisets. For example, when considering the references of a single paper, the publication dates of those references form multisets of varying size (e.g., {1995, 1995, 1996}, {1975, 1986, 1998, 1998}). The RBC assumes each value of a multiset is independently drawn from the same multinomial distribution. This approach is designed to mirror the independence assumption of the naive Bayesian classifier [4]. In addition to the conventional assumption of attribute independence, the RBC also assumes attribute value independence within each multiset.

For a class label C, attributes $\mathbf{X}$, and related items R, the RBC calculates the probability of C for an item i of type $G(i)$ as follows:

$$P(C^i|\mathbf{X}, R) \propto \prod_{X_m \in \mathbf{X}^{\mathbf{G(i)}}} P(X_m^i|C) \cdot \prod_{j \in R} \prod_{X_k \in \mathbf{X}^{\mathbf{G(j)}}} P(X_k^j|C) \cdot P(C) \tag{1}$$

In equation 1, each feature variable is implicitly weighed with a weight 1. In order to incorporate the weights from the summary graph we modify this equation as follows:

$$P(C_t^i|\mathbf{X}, R) \propto \prod_{X_m \in \mathbf{X}^{\mathbf{G(i)}}} P(X_m^i|C) \cdot \prod_{j \in R} \prod_{X_k \in \mathbf{X}^{\mathbf{G(j)}}} w_{ij}^t \cdot P(X_k^j|C) \cdot P(C) \tag{2}$$

where w_{ij}^t is the sum of the weights in the summary graph E_t^S on the path from the target node i to the related node j. For example, if paper i references paper j, then w_{ij} is the weight on the direct link between i and j. However, if paper i is authored by person j who has coauthored with person k (and k is not an author of i), then $w_{ik} = w_{ij} + w_{jk}$. The weights can be viewed as probabilities that a particular relationship in the data is still active at the current time step, given that the link was observed at time $t-k$. This is modeled with a geometric distribution using the parameter θ, which specifies the probability that a link occurred k time steps in the past.

In the experimental section below, we will refer to our approach as TV-RBC for *time-varying* RBC. The time complexity of TV-RBC learning is $O((t+m)E)$, where t is the number of time steps, m is the number of attributes used for prediction, and E is the number of edges in the summary graph G^t. The time complexity for the summary graph computation is $O(tE)$ since each edge must be considered in each summary calculation. In the worse case, when the graph is close to completely connected, the number of edges will be $O(V^2)$. However, in many relational datasets, node degree is bounded by a low constant (i.e., a node's neighbors do not grow as a function of dataset size), thus $E << V^2$ in practice. Also, if we assume that the number of neighbors for each node (used to get a particular set of attribute values) can be bounded by a low constant, then the time complexity for the RBC learning algorithm is $O(mV)$ where V is the number of target nodes (i.e., instances being predicted). However, in the worst case, as the summary graph becomes completely connected, the algorithm will be dominated by the number of edges in the graph, thus the complexity of RBC learning is $O(mE)$.

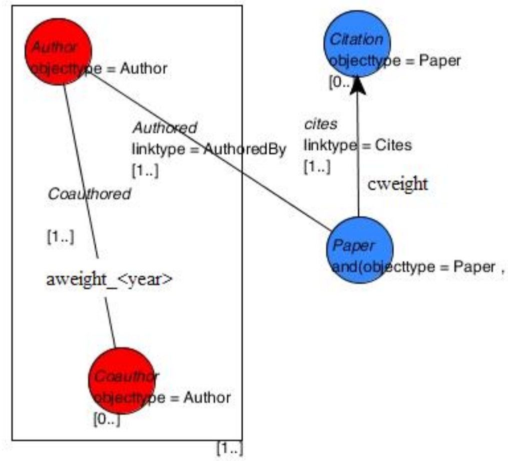

Figure 4: The attributes *cweight* and *aweight_<year>* are the weight attributes computed from the summary graphs.

4. EXPERIMENTAL EVALUATION

We report an initial evaluation of the TV-RBC on data drawn from Cora, a database of computer science research papers extracted automatically from the web using machine learning techniques [10]. We selected the set of 4,330 machine learning papers along with associated authors, cited papers, and related coauthors. We computed a summary graph G_t^S for each year $t \leq 1998$, where the links in the graphs comprise of the coauthor relations between people and the citation relations between papers.

We evaluated the models in a classification context to assess the impact of our weighting scheme on model performance. We considered a set of temporal samples, one for each year between 1993 and 1998, learning the models on one year and then testing the models on the sample from the subsequent year. The classification task was to predict one of seven machine-learning paper topics (e.g., *Genetic Algorithms*). The attributes supplied to the model were *Citation.topic* and *Coauthor.most-prevalent-topic*. Figure 4 shows the query we used to identify the relational neighborhood of a paper. The query matches all research papers in a given sample and returns for each paper a subgraph that includes all authors and references, and coauthors associated with the paper.

We compared two models. The first is a baseline (snapshot) RBC model that uses a graph $G_{\leq t}$, which consists of all links up to and including the year t of the sample. The RBC approach uses equation 1 and does not weight the attributes or the links. The second model is the TV-RBC, which first summarizes the graph into G_t^S using all years up to and including t and then weights the attributes appropriately during learning and inference according to equation 2. We evaluated the models using area under the ROC curve (AUC).

The first set of results are plotted in Figures 5,6 and 7. Here we compare the baseline RBC the TV-RBC with a fixed $\theta = 0.7$. Figure 5 plots the variation of AUC across different years for TV-RBC as well as the baseline RBC approach. Our approach results in an 8-12% improvement over the baseline RBC model. We used a two-tailed, paired t-test to assess the significance of the AUC results obtained from the five temporal splits. The t-test compares TV-RBC to baseline-RBC, with a null hypothesis of no difference in AUC. The improvement is significant with $p < 0.001$.

Next we compared the two models with ablated data to assess the temporal content in each type of link. Figure 6 plots the variation of AUC across different years using only the coauthor attributes and links for classification. Again, the improvement in AUC is significant at the $p < 0.001$ level. However, without the topic attributes of the references, the AUC values are lower than those in Figure 5.

Similarly, Figure 7 plots the variation of AUC but this time the model uses only the attributes and links of the references. Again, the improvement in AUC is significant at the $p < 0.001$ level. In all three cases, the TV-RBC performs substantially better than baseline RBC model. Furthermore, we also confirm that we get better prediction accuracies using both the citation and coauthor attributes than either alone.

The value of the summary parameter θ that we have used in the above experiments was fixed to be 0.7 which means our summary function weighed past data points with 30% weight and the present data points with 70% weight. We varied θ across different values to see what is the effect of the summary parameter on TV-RBC performance. These results are graphed in Figures 8, 9 and 10 on the full and ablated data. Ideally, we would expect it to be an inverted parabolic curve peaking at some suitable value of θ between 0 and 1.

Figures 8, 9 and 10 plot the variation of AUC with θ across each of the temporal samples. As expected, the curve resembles an inverted parabola peaking in the neighborhood of 0.7. In other words, learning and prediction over sum-

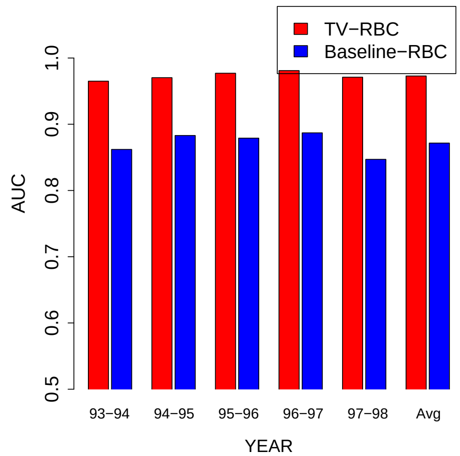

Figure 5: AUC evaluation on Cora data.

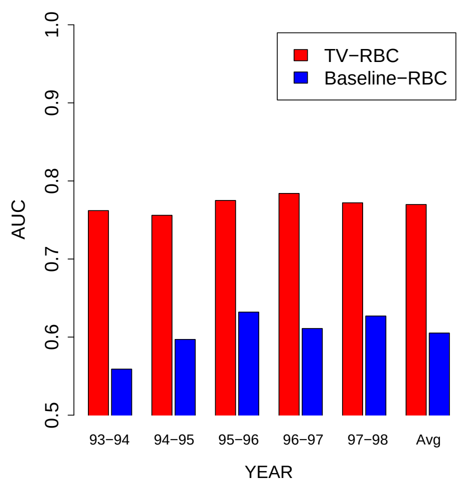

Figure 6: AUC evaluation on Cora data using only coauthor attributes/links.

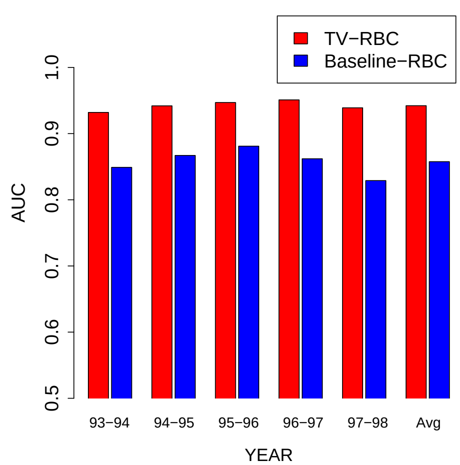

Figure 7: AUC evaluation on Cora data using only reference attributes/links.

marized, weighted attributes works better than using just the present information ($\theta = 1.0$). It also works better than using too much past information ($\theta = 0.25$). Note that the TV-RBC model outperforms the snapshot RBC model for all reported values of θ as well. The maximum performance of the baseline model is 0.887 on the 1996-97 sample. This is less than the minimum TV-RBC performance of 0.913 on the 1993-94 sample with $\theta = 0.25$.

The difference in AUC values when considering only the coauthor attributes and when considering only the reference attributes is evident again from these figures. The model performs best when both the attributes are considered but reference attributes seem to have a higher influence on the prediction task than the coauthor attributes which is expected considering the distance of coauthor attributes from the paper objects in our relational schema.

5. CONCLUSIONS AND FUTURE WORK

This paper presents a new approach to modeling relational data with time-varying link structure. To date, work on statistical relational models has been primarily on static snapshots of relational datasets even though most relational domains have temporal dynamics that are important to model. Although there has been some work modeling domains with time-varying attributes, to our knowledge this is the first model that exploits information in dynamic relationships between entities to improve prediction. This work has demonstrated that significant performance gains can be achieved by incorporating temporal-relational information into statistical relational models, even in a simple weighted summarization approach.

We evaluated the algorithm in a citation domain, but the algorithm is applicable to a wide array of relational domains where the relationships between entities change over time. For example, our algorithm can be easily applied to predict the content category of a news magazine based on the references it makes. We can also predict the success of a movie given the history of the star cast of the movie and the relationships between them in the past (e.g. co-starred). Also, our method of computing a weighted-summary graph can be combined with any statistical relational learner. We have chosen to use RBCs in this work due to their simplicity and to avoid any confounding effects of feature selection. However, future work will explore the performance of alternative predictive models (e.g., [17, 6, 19]).

There are a number of ways to improve on this work. First, we intend to develop a temporal cross-validation approach to set the value of θ automatically during learning. Next, we plan to evaluate the algorithm on a number of other real-world domains, using ablation studies to identify the link types and parameter settings that offer the most improvement in accuracy. Finally, we plan to extend the approach to model temporally-varying attributes as well—in this case the weighted summarization would produce distributions of attribute values for each node.

One of the strengths of our approach is that it is a relatively simple and efficient way of incorporating time into statistical relational models. While our algorithm doesn't make a Markov assumption about the temporal dependencies, it is

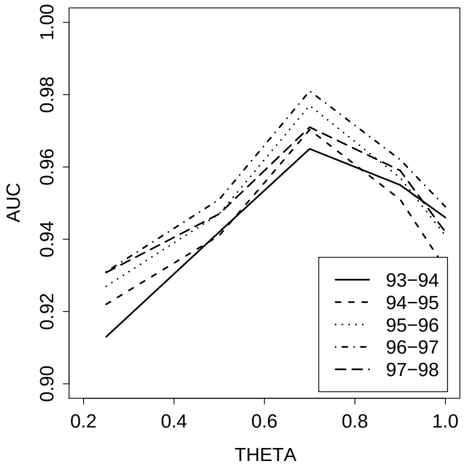

Figure 8: TV-RBC performance as θ is varied.

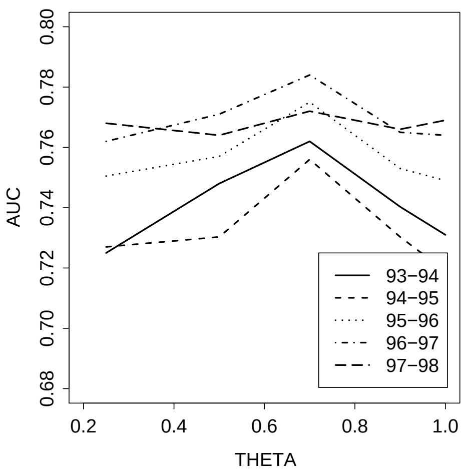

Figure 9: TV-RBC performance as θ is varied, using only coauthor attributes/links.

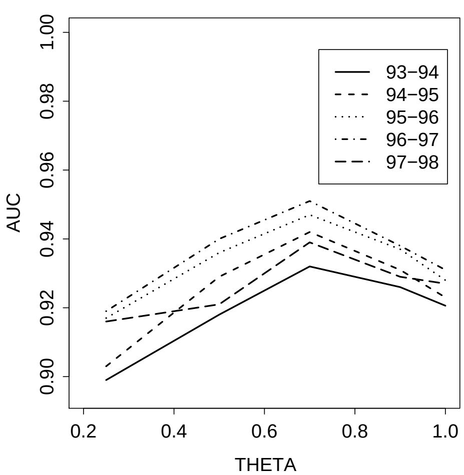

Figure 10: TV-RBC performance as θ is varied, using only reference attributes/links.

predicated on the assumption that events in the recent past are more informative than events in the distant past. It also assumes that all relationships are meaningful (i.e., all edges are included in the summary graph calculation). A full joint *temporal-relational* model may be able to represent the dependencies in the data more accurately, however without a means to limit either the temporal or relational dependencies, the dimensionality of the cross-product model will be far too large for accurate estimation with finite datasets.

This work attempts to model temporal dependencies by specifying a limited number of temporal patterns to moderate the relational dependencies. Other efforts to identify and exploit temporal *motifs* for use as relational features may be a promising means to extend the relational model space in a restricted way while still capturing most of the relevant temporal information in the data. Such efficient methods for temporal-relational data mining will be invaluable to a number of relational domains including citation analysis, fraud detection, and homeland security.

6. REFERENCES

[1] D. Cai, Z. Shao, X. He, X. Yan, and J. Han. Community mining from multi-relational networks. In *Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases*, 2005.

[2] S. Chakrabarti, B. Dom, and P. Indyk. Enhanced hypertext categorization using hyperlinks. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 307–318, 1998.

[3] C. Cortes, D. Pregibon, and C. Volinsky. Communities of interest. In *Proceedings of the 4th International Symposium of Intelligent Data Analysis*, pages 105–114, 2001.

[4] P. Domingos and M. Pazzani. On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29:103–130, 1997.

[5] P. Domingos and M. Richardson. Mining the network value of customers. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 57–66, 2001.

[6] N. Friedman, L. Getoor, D. Koller, and A. Pfeffer. Learning probabilistic relational models. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence*, pages 1300–1309, 1999.

[7] Z. Ghahramani. Learning dynamic Bayesian networks. In *Adaptive Processing of Sequences and Data Structures*, pages 168–197. Springer-Verlag, 1998.

[8] S. Hill, D. Agarwal, R. Bell, and C. Volinsky. Building an effective representation of dynamic networks. *Journal of Computational and Graphical Statistics*, Sept, 2006.

[9] D. Jensen, J. Neville, and B. Gallagher. Why collective inference improves relational classification. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 593–598, 2004.

[10] A. McCallum, K. Nigam, J. Rennie, and K. Seymore. A machine learning approach to building domain-specific search engines. In *Proceedings of the*

16th International Joint Conference on Artificial Intelligence, pages 662–667, 1999.

[11] A. McGovern, L. Friedland, M. Hay, B. Gallagher, A. Fast, J. Neville, and D. Jensen. Exploiting relational structure to understand publication patterns in high-energy physics. *SIGKDD Explorations*, 5(2):165–172, 2003.

[12] M. McPherson, L. Smith-Lovin, and J. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27:415–445, 2001.

[13] B. Milch, B. Marthi, S. Russell, D. Sontag, D. Ong, and A. Kolobov. Blog: Probabilistic models with unknown objects. In *Introduction to Statistical Relational Learning*, 2007.

[14] J. Neville, O. Şimşek, D. Jensen, J. Komoroske, K. Palmer, and H. Goldberg. Using relational knowledge discovery to prevent securities fraud. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 449–458, 2005.

[15] J. Neville, D. Jensen, L. Friedland, and M. Hay. Learning relational probability trees. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 625–630, 2003.

[16] J. Neville, D. Jensen, and B. Gallagher. Simple estimators for relational Bayesian classifers. In *Proceedings of the 3rd IEEE International Conference on Data Mining*, pages 609–612, 2003.

[17] C. Perlich and F. Provost. Aggregation-based feature invention and relational concept classes. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 167–176, 2003.

[18] S. Sanghai, P. Domingos, and D. Weld. Learning statistical models of time-varying relational data. In *Proceedings of the Workshop on Statistical Relational Learning, 18th International Joint Conference on Artificial Intelligence*, 2003.

[19] B. Taskar, P. Abbeel, and D. Koller. Discriminative probabilistic models for relational data. In *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence*, pages 485–492, 2002.