Increasing the Predictive Power of Affiliation Networks

Lisa Singh
Computer Science Dept.
Georgetown University
Washington, DC, USA
singh@cs.georgetown.edu

Lise Getoor
Computer Science Dept.
University of Maryland
College Park, MD, USA
getoor@cs.umd.edu

Abstract

Scale is often an issue when attempting to understand and analyze large social networks. As the size of the network increases, it is harder to make sense of the network, and it is computationally costly to manipulate and maintain. Here we investigate methods for pruning social networks by determining the most relevant relationships in a social network. We measure importance in terms of predictive accuracy on a set of target attributes of social network groups. Our goal is to create a pruned network that models the most informative affiliations and relationships. We present methods for pruning networks based on both structural properties and descriptive attributes. These pruning approaches can be used to decrease the expense of constructing social networks for analysis by reducing the number of relationships that need to be investigated and as a data reduction approach for approximating larger graphs or visualizing large graphs. We demonstrate our method on a network of NASDAQ and NYSE business executives and on a bibliographic network describing publications and authors and show that structural and descriptive pruning increase the predictive power of affiliation networks when compared to random pruning.

1 Introduction

A social network describes a set of actors (e.g., persons, organizations) linked together by a set of social relationships (e.g., friendship, transfer of funds, overlapping membership). Social networks are commonly represented as a graph, in which the nodes represent actors and edges represent relationships. Examples of social networks include online communication networks, disease transmission networks, and bibliographic citation networks. There is a growing interest in methods for understanding, mining, and discovering predictive patterns in social networks.

An *affiliation network* is a special kind of social network in which there are two kinds of entities, actors and events, and there is a participation relationship which relates them. Affiliation networks are commonly represented as bipartite graphs, in which there are two kinds of nodes, representing actors and events, and edges link actors to events. Examples of affiliation networks include: 1) corporate board memberships, where the actors are executives, the events correspond to different company boards, and the links indicate which executives serve on which company boards; 2) author collaboration networks, where the actors are authors, the events are

Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

papers, and the links indicate co-authors of papers; and 3) congressional voting records, where the actors are the congressional members, the events are the bills, and the links represent the supporters for a bill.

A social network has both structural properties and descriptive attributes. The structural properties are determined by the graph structure of the network. Examples include the density of the graph, the average degree of nodes in the graph, the geodesic distance in the graph, the number of cliques in the graph, etc. In addition to structural properties, actors, events and relationships often have associated descriptive attributes containing features specific to the social context of the network. These are typically represented as attributes of the nodes or edges. For example, a corporate board social network may contain descriptive attributes representing the job function and age of a board member. A disease transmission social network may contain descriptive attributes representing the location of person's home and date of disease discovery.

Recent literature in the network science community has focused on understanding the structural properties of social networks and the construction of models for generating networks which have certain structure characteristics (degree distribution, small-world effects, etc.). Computer scientists are mining social networks based on these structural properties of networks. However, developing methods which combine network structure and descriptive attributes are necessary for accurate predictive modeling.

Predictive modeling can also be used to study approaches for compressing the representation of a social network, while maintaining its predictive accuracy. In the past, the social networks as studied in sociology tended to be relatively small, often with only tens of nodes. However, given the great increase in ability to both gather and process data, the social networks being analyzed today can be quite large. Because the data used to describe the network may not originally have been collected for the purpose of social network analysis, the data may contain irrelevant, redundant or noisy information. Noisy and redundant information can make networks difficult to interpret. Automatic techniques for identifying relevant aspects of the social networks can help improve computational efficiency and may at the same time improve understandability. Furthermore, since recording changes to a social network and maintaining consistency can be expensive, some applications can benefit from minimizing the amount of information stored.

In this paper, we begin by giving an overview of some of the representational issues related to social networks, especially affiliation networks. Next, we describe different pruning strategies for social networks. Our aim is to find compressed networks that maintain predictive and descriptive quality. Here we measure the compression in terms of the description length of the network and we measure the quality by measuring the predictive accuracy for the event attribute classifier built from the compressed network. We have evaluated our pruning methods on two real-world data sets. One is a network of NASDAQ and NYSE business firm executives and board members. The second is a bibliographic network describing publications and authors. We have found that we can achieve significant compression without sacrificing (and in some cases improving) predictive accuracy. This paper extends the work introduced in [17].

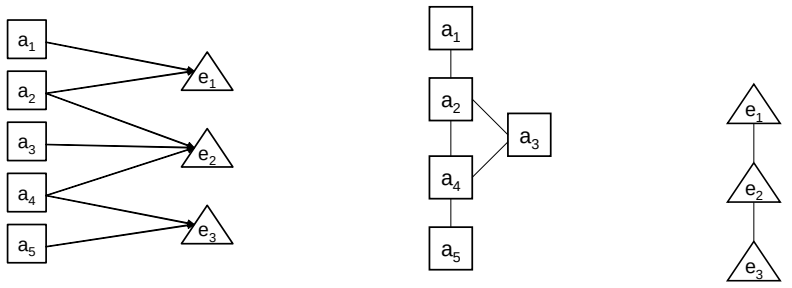
2 Affiliation Networks

Definition 1: An *affiliation network* N consists of a set of actors A, linked via a set of relationships R to a set of events E, $N = A, R, E$, where

$$\begin{aligned} A &= \{a_1, \ldots, a_n\}, \\ E &= \{e_1, \ldots, e_m\}, \\ R &= \{r_{ij}\}, \text{ where } r_{ij} \text{ denotes actor } a_i \text{ participates in event } e_j, \end{aligned}$$

and n is the number of actors and m is the number of events.

An affiliation network may be represented using many different graph structures. The most common representation for affiliation networks is as a bipartite graph, which we will call an *actor-event* graph, AE. In



(a) (b) (c)

Figure 1: (a) A simple affiliation network with actors a_1, a_2, a_3, a_4 and a_5 and events e_1, e_2 and e_3 (b) The co-membership graph for the affiliation network (c) The event overlap graph for the network.

this representation, there are two different node types representing actors and events. Networks with two node types are called *two-mode* or *bi-modal*. Figure 1(a) shows a small example of a two-mode actor-event node graph. The squares in the figure represent actors and the triangles represent events. The membership relations are highlighted in this graph structure.

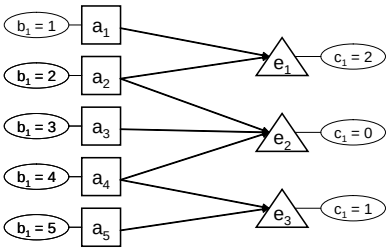
There are several useful projections of the actor-event graph. To focus on actors, one can perform a unipartite projection of the actors on the two-mode affiliation graph. The resulting network is a *single-mode* or *uni-modal* network, where we have a single object type and a single edge type. Representing an affiliation network in this way results in what is referred to as the *co-membership* graph, CM. The co-membership graph has a node for each actor, and an edge between actors who participate in the same event. Similarly, to focus on events, one can projection the actor-event graph onto the events. This results in what is called an *event overlap* graph, EO. It also contains a single node type and a single edge type. In the event overlap graph, the emphasis is on the connections among events. This graph has a node for each event, and an edge between events that share a common actor. Figure 1(b) shows the co-membership graph corresponding to the actor-event graph in Figure 1(a), and Figure 1(c) shows the event overlap graph corresponding to the same actor-event graph.

In addition to the nodes and edges themselves, the nodes and edges in the affiliation network can have descriptive attributes or features associated with them. Figure 2(a) shows the affiliation graph along with descriptive attributes for the actors and events (shown in ovals). In a corporate board social network, executives may have attributes such as education level, academic degree and age, companies may have attributes describing the corporation such as industry, sector, stock exchange and share price, and the serves-on-board relation may have attributes describing the relationship between the corporation and the executive such as position on the board and length of tenure on the board.

It is straight-forward to represent an affiliation network in relational algebra. We introduce the relations $A(Id_A, B_1, \ldots, B_k)$,$E(Id_E, C_1, \ldots, C_l)$, and $R(Id_A, Id_E, D_1, \ldots, D_m)$, representing the actors, the events, and the participation relations of a network. Here the Id_A, Id_E, and (Id_A, Id_E) are primary keys and the B_i, C_j and D_k are descriptive attributes for the relations A, E and R, respectively.

3 Prediction in Social Networks

Our goal is to develop principled approaches to compressing and pruning social networks. Our approach is to determine which portions of the network can be removed while minimizing information loss. Let $N = (A, E, R)$ be ther original network and $N' = (A', E', R')$ be the pruned network (we will describe how we construct the pruned network shortly). We begin by describing the predictive accuracy measure used to assess the performance



(a)

Id_E	c_1	Max(b_1)	Avg(c_1)
1	2	2	0
2	0	4	1.5
3	1	5	0

(b)

Figure 2: (a) The affiliation graph with descriptive attributes for the actors and events shown in ovals. (b) The constructed attributes for the events.

of different pruning approaches.

Here, we will focus on maximizing our predictive accuracy on the event attributes. For ease of exposition, we will assume we are attempting to maximize the predictive accuracy for a single event attribute $E.C_i$, based on attributes of related actors found using the co-membership information and based on attributes of related events found using the event overlap information. The idea is to construct a classifier, using local neighborhood information, to predict $E.C_i$. Now it is easy to see the difficulty with this setup. Each event may have a different number of related actors and a different number of related events, so how can we construct features to use in our classifier?

We solve this problem by computing an aggregate over the set of related actors and over the set of events. Aggregation is a common technique used to construct feature vectors in relational domains [11, 15]. Here we assume some set of aggregates is associated with each attribute. For the actor attributes $\{B_1, \ldots, B_k\}$, we have associated aggregate operators $\{a_{B_1}, \ldots a_{B_k}\}$ and for the event attributes $\{C_1, \ldots, C_l\}$, we have associated aggregate operators $\{a_{C_1}, \ldots, a_{C_l}\}$,

We begin by computing the aggregates over the set of related actors:[1]

$$AA(Id_E, A_{B_1}, \ldots, A_{B_k}) = \gamma_{Id_E, a_{B_1}(B_1), \ldots, a_{B_k}(B_k)}(R \underset{R.Id_A = A.Id_A}{\bowtie} A)$$

which we call AA for aggregates over actors. Next we compute aggregates constructed from the related events:

$$AE(Id_E, A_{C_1}, \ldots, A_{C_l}) = \gamma_{Id_E, a_{C_1}(C_1), \ldots, a_{C_l}(C_l)}(EO \underset{EO.R.Id_E = E.Id_E}{\bowtie} E)$$

which we call AE for aggregates over events.

We can combine these relations with the event relation E to create a set P_E containing both constructed features and event attributes.

$$P_E = E \underset{E.Id_E = AA.Id_E}{\bowtie} AA \underset{E.Id_E = AE.Id_E}{\bowtie} AE$$

[1]Recall that γ is the grouping operator in relational algebra [6].

We will use the constructed features to predict event attributes.

Example: Consider the affiliation network with descriptive attributes shown in Figure 2(a). Suppose that the aggregate operator that we use for B_1 is maximum and the aggregate operator that we use for C_1 is average. The constructed table showing the aggregates that will be used to build our classifier is shown in Figure 2(b).

The above describes in a generic way how we find the features from which we will predict event attributes. In order to actually make a prediction, we will need to first learn a classifier. Here we do not do anything out of the ordinary; we construct an appropriate training set from an observed social network. The constructed training set can be used by any supervised learning method to learn a classifier F, which predicts the value of E.C based on $\{A_{B_1}, \ldots, A_{B_k}, A_{C_1}, \ldots, A_{C_l}\}$.

We compare the classifier F_N constructed from the original social network $N = \{A, E, R\}$ with the classifier $F_{N'}$ constructed from a pruned social network $N' = \{A', E', R'\}$. We compare both accuracy on the training sets and, more importantly, accuracy on test sets. Accuracy on the training set measures how well the classifiers are able to fit the existing network. Accuracy on the test set measures how well the classifiers are able to generalize. Our goal is to find pruned networks that are both compact and accurate on both sets.

4 Pruning Techniques

Next we describe different pruning strategies. We consider two categories of operations. The first involves removing edges from the affiliation network. The second involves removing actors (and incident edges) from the affiliation network. We can use different techniques for pruning a network. The three techniques of interest to us are: 1) pruning based on structural properties, 2) pruning based on descriptive attribute values, and 3) pruning based on random sampling.

Structural Pruning Structural network properties or measurements involve evaluating the location of actors in a social network. Measuring the network location involves finding the centrality of a node. Structural measures have traditionally been used to identify prominent or important nodes in a social network. Two well known centrality measures are *degree* and *betweenness*. The degree of a node is defined as the number of direct connections a node has to other nodes in the network. The nodes with the most connections are considered the most active nodes in the network. They are referred to as the connectors or the *hubs* in the network. Betweenness of a node corresponds to the number of shortest paths going through the node. Nodes with high betweenness are referred to as *brokers*. A variation of this that is appropriate for affiliation networks is the number of cliques a node connects. This allows us to identify nodes that connect one group of actors to another group of actors. In traditional uni-mode networks, this could be a node that links two clusters that it does not participate in. It acts as a bridge between these clusters. In affiliation networks, this measure identifies nodes that participate concurrently in multiple events. These brokers are boundary spanners that have access to information flow in multiple clusters. They tend to have great influence in the network [19].

Therefore, when pruning based on structure, we will be interested in removing actors that are not hubs and/or brokers from the network.

Descriptive Attribute-based Pruning Another pruning technique of interest involves pruning based on descriptive attributes. We prune edges by selecting on attributes D_j of the R relation,

$$R' = \sigma_{R.D_j = d_j}(R),$$

where d_j is some constant attribute value. In other words, we will remove edges from our graph based on values for D_j. We look at both the case where we keep *only* edges with value d_j for D_j, and also the case where we keep all edges *except* edges with value d_j. Pruning edges may result in pruning both actor and event nodes if after pruning there are no edges connecting them to the network.

In addition, we prune actors by selecting on attributes B_j of actor relation A,

$$A' = \sigma_{A.B_j=b_j}(A),$$

where b_j is some constant attribute value. Pruning actors also results in a reduction in the number of edges, since we drop any edges to non-selected actors.

Random Sampling Finally, as a baseline, we compare pruning based on random sampling. This involves maintaining only a random sample of the actor population for analysis. Random sampling is a traditional statistical approach to approximating large graph structures.

Compression It is important to quantify the compression achieved by pruning. We use a relatively generic measure, the description length of the graph,

$$DL(N) = \log(|A|) + \log(|E|) + |R|(\log(|A|)\log(|E|))$$

where the logs are base two. $DL(N)$ is the number of bits required to represent the network. We need the first two terms to describe the number of actors and the number of events and the final term is the number of bits required to represent the edges.

5 Experimental Results

In this section we evaluate the degree of compression and the predictive accuracy of different pruning approaches.

5.1 Data Sets

We analyzed two affiliation networks. The first data set, the Executive Corporation Network (ECN), contains information about executives of companies that are traded on the NASDAQ and the NYSE. The executives serve on the Board of Directors for one or more of the companies in the data set. This data was collected from the Reuter's market data website (yahoo.mulexinvestor.com) in January 2004. There are 66,134 executives and 5384 companies (3284 NASDAQ and 2100 NYSE). The executives are the actors in the ECN, the companies are the events and board membership is the connecting relationship between the actor nodes and the event nodes. The relational schema is:

- A = Executive(exec_id, exec_name, age, education_level)
- E = Company(co_id, co_name, stock_exchange, sector, stock_price)
- R = BoardMembership(exec_id, co_id, officer_position, join_date)

The average board size is 14, the average number of boards an officer is on is 1.14, the number of officers serving on multiple boards is 6544, and the average number of boards these officer are on is 2.4. We attempt predicting two attributes, *stock_exchange* and *sector*. A sector is a coarse grouping of industries of the companies, e.g., telecommunications and health care. When pruning on descriptive attributes, we consider attributes of both the Executive relation and the BoardMembership relation. One example is *officer_position*, e.g., CEO, President, Treasurer and Director.

The second data set, the Author Publication Network (APN), contains information about publications and their authors. This data set was created using a portion of the ACM SIGMOD anthology in 2004. We focused on a subset of the periodicals and authors where there was at least one reference to the publication. In the final data set we analyzed, there were 13,070 authors and 16,287 publications.

The authors are the actors in the APN and the publications are the events. Paper authorship is the connecting relationship. The relational schema is:

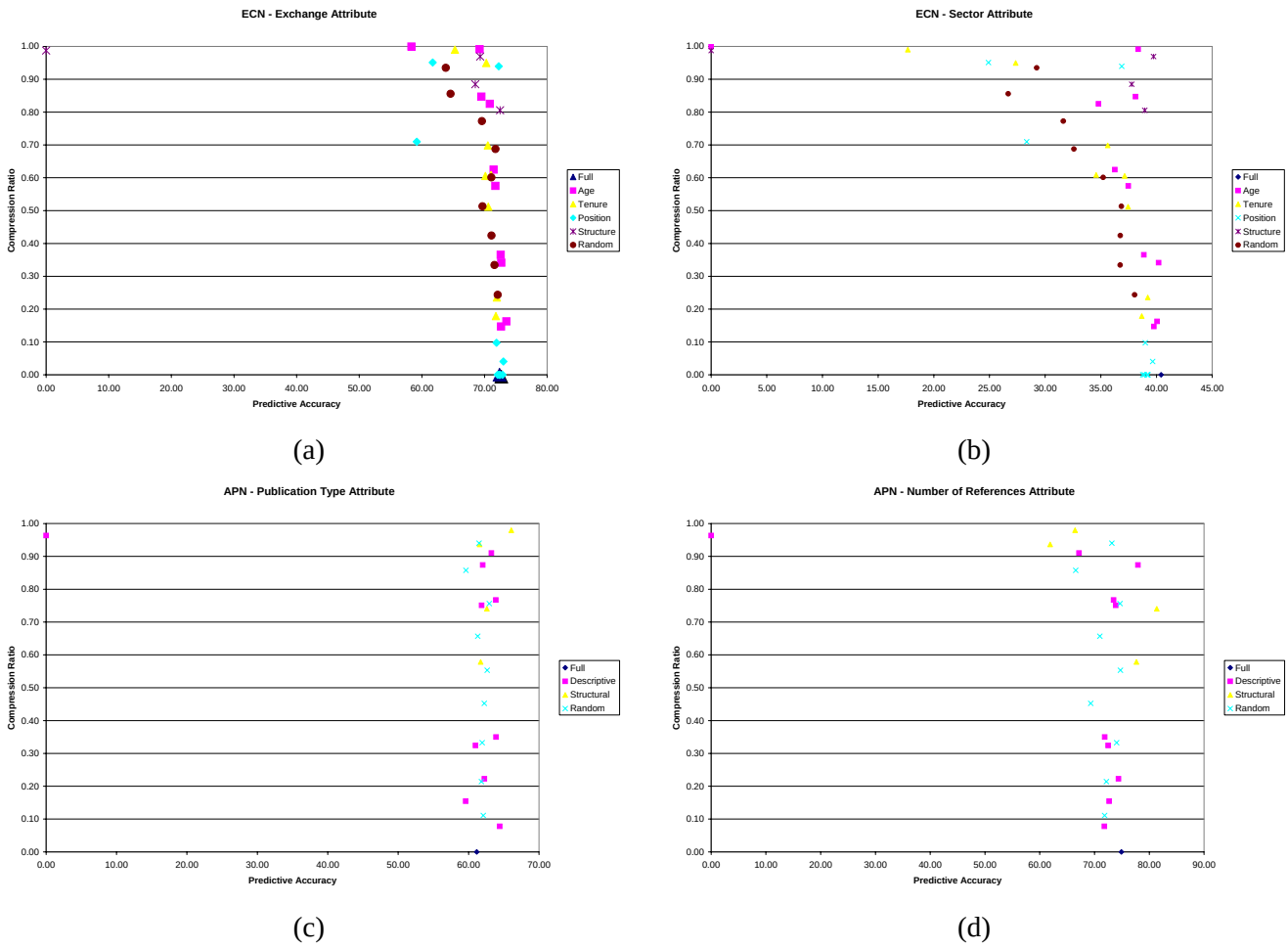


Figure 3: Comparisons of compression vs. accuracy for a variety of network pruning strategies for a) ECN exchange b) ECN sector c) APN pulbication type and d) APN number of references.

- A = Author(author_id, author_name, affiliation, number_of_publications)
- E = Publication(pub_id, pub_type, pub_date, number_of_references, number_of_citations)
- R = PaperAuthorship(author_id, pub_id)

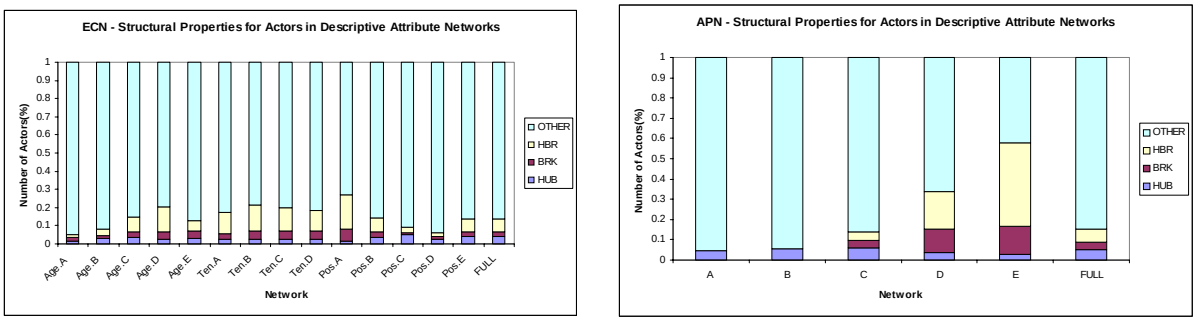
The average number of authors per publication is 2.4 and the average number of publications per author is 2.9. For APN, we predicted the two event attributes *pub_type* and *number_of_references* (to publication).

5.2 Accuracy and Compression Results

Our goal is to find small networks that can accurately predict event attributes. We compare the following affiliation networks:

- no pruning (**full**)
- descriptive attribute pruning (**descriptive**)
- pruning based on hubs and/or brokers (**structural**)
- random sampling (**random**)

We built event-attribute classifiers from the networks as described in Section 3. For categorical aggregate attributes, we calculated the mode of the neighboring event values, and for numeric aggregate attributes, we



(a) (b)

Figure 4: The structural characteristics of actors in different prunings for a) ECN and b) APN.

calculated the minimum, maximum and average of the neighboring event values. Once the predictive models have been generated, we evaluate the predictive accuracy of the complete network and the different pruned networks. We also compare the compression rations in terms of descriptive length, $DL(G)$. The classifiers were then created using WEKA. We tested a range of classification algorithms including decision trees, Naive Bayes, and support vector machines (SVMs). The results were relatively consistent across classifiers; due to space constraints, here we present results only for SVMs using five-fold cross validation.

When constructing our feature vector, we constructed aggregates for the following ECN actor and event attributes: stock exchange, industry, sector, number of officers on a board, number of advanced degrees on a board and officer age of a board. We evaluated three descriptive prunings. The first two descriptive prunings, *position* and *tenure*, involve removing edges from our affiliation graph for executives based on the attributes *BoardMembership.officer_position* and *BoardMembership.join_date*. For example, one pruning of *BoardMembership.officer_position* keeps only edges of CEOs and removes all other membership edges from the network. The third descriptive pruning involves removing actors based on age.

To group attribute values, we binned numeric attributes and we abstracted categorical attributes. Binning for each descriptive attribute used for pruning was created based on maintaining approximate equal size buckets or based on sematically interpretable abstractions. For both our networks, the binnings resulted in four to five bins for each attribute. For example, the attribute bands for *BoardMembership.officer_position* are as follows:

- A - Chairman of the Board
- B - Executive Officer (CEO, President, COO, etc.)
- C - Senior Officer (VP, Sr. VP, Comptroller, etc.)
- D - Board Officer (Treasurer, Secretary, etc.)
- E - Director

For the APN, we used the attribute *Author.number_of_publications* for descriptive pruning.

As mentioned earlier, descriptive attribute pruning has one of two interpretations for an attribute B with attribute value c: 1) maintain *only* actors with $B = c$ (**only**) and 2) maintain all actors *except* where $B = c$ (**except**). We evaluated pruning on every descriptive attribute value for each descriptive pruning category.

For structural pruning, we tested four cases: maintaining only actors who are hubs, (**HUB**), maintaining only actors who are brokers, (**BRK**), maintaining only actors who are both hubs and brokers, (**BOTH**), and maintaining only actors who are hubs or brokers. (**HBK**). Finally, for random pruning, we compared results on random samples for 9 different sample sizes (between 10% and 90% of the actors in the network).

Figure 3 shows compression versus predictive accuracy for two different attributes in each data set. The right upper corner represents the 'best' networks in terms of compression and predictive accuracy. Figure 3(a) shows results for predicting the ECN *exchange* attribute. The classifier built using the full network achieves an accuracy of 72.4%. The best accuracy and compressions are for networks pruned using descriptive pruning.

Pruning on position, we achieve an accuracy of 72.3% with a compression of 94%. In this case, we removed all actors except for the chairs of the company boards. Pruning on tenure, we achieve an accuracy of 70.29% with a compression of 95%, and pruning on age, we achieve an accuracy of 69.2% with a compression of 99%. In this case, we kept only the older executives. These accuracies are all better than the baseline prediction accuracy of 61% achieved by simply choosing the most common exchange.

For predicting the ECN sector, shown in Figure 3(b), the full network achieves accuracy of 40.4%. Here pruning based on both descriptive and structural properties perform well. When pruning based on age, we achieve accuracy of 40.2% with compression of 34%. In this case we kept the younger executives rather than the older ones. When pruning based on structure, we achieve accuracy of 39.7% and compression of 97% by keeping only the brokers. Figure 3(c) and (d) show similar results for the pruned APN networks, with many of the pruned networks achieving significantly higher accuracies than classifiers built from the full network. For both APN attributes, the network pruned on structure that achieved the best accuracy-compression tradeoff was the one that kept only the actors that were both hubs and brokers.

For both data sets, pruning on descriptive attributes and structure properties outperformed random pruning. One question this raised was whether or not the different pruning techniques were removing the same nodes and edges or different ones? To address the first question, Figure 4 shows the percentage of structural actor types (hubs, brokers (BRK), hubs and brokers (HBR), and other) preserved under various descriptive pruning strategies. These graphs show that for both data sets, the networks created using descriptive pruning contain a different mix of actors than those created using structural pruning. This supports our claim that structural pruning and descriptive pruning are two distinct methods for compressing networks and maintaining information rich nodes for prediction in affiliation networks.

6 Related Work

A large portion of the work in mining social networks has focused on analyzing structural properties of the networks. For a recent survey, see Newman [13]. Much of the work has been descriptive in nature, but recently there has been more work which uses structural properties for prediction. Within this category, a number of papers focus on the spread of influence through the network (e.g., [5, 9, 3]). These papers attempt to identify the most influential nodes in the network. Domingos and Richardson [5] use a global, probabilistic model that employs the joint distribution of the behavior over all the nodes. Kempe et al. [9] use a diffusion process that begins with an initial set of active nodes and uses different weighting schemes to determine whether or not a neighbor should be activated. Liben-Nowell and Kleinberg [12] attempt to predict future interactions between actors using the network topology. In addition, Palmer et al. [14] propose an efficient method for approximating the connectivity properties of a graph.

Other work uses structural properties for both classification and clustering. Agrawal et al. [1] use the link structure of newsgroup social networks to classify user behavior within a newsgroup, specifically they identify whether a respondent agrees with a posting. Schwartz and Wood [16] create an email graph with edges corresponding to sets of shared interests and present an algorithm that analyzes the graph structure to cluster users with similar interests. Their approach derives a specialization subgraph from the relationship clusters.

Graph sampling and compression is also a relevant, active area of study. As we saw in section 5, random sampling did not generally lead to good prediction results. This finding agrees with that of Airoldi and Carley [2]. They find that pure network topologies are sensitive to random sampling. As mentioned earlier, graphs have been compressed using different local network measures [4]. A similar approach is to use frequently occurring subgraphs as proposed in [10].

There is also a related line of work which makes use of the descriptive attributes of the entities in the network for collective classification (e.g., [8, 18, 7]). While potentially applicable here as well, our focus is not on collective classification.

7 Conclusions

Exploring descriptive and structural pruning techniques together is needed for compact and accurate compression of networks. In this paper we showed how to use structural properties and descriptive attributes to prune social networks. We began by introducing a general framework for representing affiliation networks using relational algebra to formally express different network representations. We then used relational algebra expressions to define pruning strategies based on structural properties and descriptive attributes. Finally, we demonstrated the effectiveness of these pruning approaches on two real world data sets. While the networks resulting from structural pruning and descriptive pruning are quite distinct, both are viable approaches for reducing the size of a social network while still maintaining predictive accuracy on a set of target event attributes. Both approaches perform better than random sampling and lead to understandable, compressed networks that maintain (and in some cases increase) predict power.

References

[1] R. Agrawal, S. Rajagopalan, R. Srikant, and Y. Xu. Mining newsgroups using networks arising from social behavior. In *International World Wide Web Conference*, 2003.

[2] E. M. Airoldi and K. M. Carley. Sampling algorithms for pure network topologies: a study on the stability and the separability of metric embeddings. *SIGKDD Explorations Newsletter*, 7(2):13–22, 2005.

[3] M. Boguna and R. Pastor-Satorras. Epidemic spreading in correlated complex networks. *Physical review*, E 66(4), 2002.

[4] N. Deo and B. Litow. A structural approach to graph compression, 1998.

[5] P. Domingos and M. Richadson. Mining the network value of customers. In *ACM Intl. Conf. on Knowledge Discovery and Data Mining*, 2001.

[6] H. Garcia-Molina, J. Ullman, and J. Widom. *Database Systems*. Prentice Hall, New Jersey, 2002.

[7] L. Getoor. Link-based classification. In S. Bandyopadhyay, U. Maulik, L. Holder, and D. Cook, editors, *Advanced Methods for Knowledge Discovery from Complex Data*. Springer, 2005.

[8] D. Jensen and J. Neville. Data mining in social networks. In *National Academy of Sciences Symposium on Dynamic Social Network Analysis*, 2002.

[9] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *ACM Intl. Conf. on Knowledge Discovery and Data Mining*, 2003.

[10] N. S. Ketkar, L. B. Holder, and D. J. Cook. Subdue: compression-based frequent pattern discovery in graph data. In *OSDM '05: Proceedings of the 1st international workshop on open source data mining*, pages 71–76, New York, NY, USA, 2005. ACM Press.

[11] A. J. Knobbe, M. de Haas, and A. Siebes. Propositionalisation and aggregates. In *Eur. Conf. on Principles of Data Mining and Knowledge Discovery*. Springer-Verlag, 2001.

[12] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *Intl. Conf. on Information and Knowledge Management*, 2003.

[13] M. Newman. The structure and function of complex networks. *IAM Review*, 45(2):167–256, 2003.

[14] C. Palmer, P. Gibbons, and C. Faloutsos. ANF: A fast and scalable tool for data mining in massive graphs. In *ACM Intl. Conf. on Knowlege Discovery and Data Mining*, 2002.

[15] C. Perlich and F. Provost. Aggregation-based feature invention and relational concept classes. In *Intl. Conf. on Knowledge Discovery and Data Mining*, 2003.

[16] M. F. Schwartz and D. C. Wood. Discovering shared interests using graph analysis. *Communications of the ACM*, 36(8), 1993.

[17] L. Singh, L. Getoor, and L. Licamele. Pruning social networks using structural properties and descriptive attributes. In *IEEE International Conference on Data Mining*, pages 773–776, Washington, DC, USA, 2005. IEEE Computer Society.

[18] B. Taskar, E. Segal, and D. Koller. Probabilistic classification and clustering in relational data. In *Intl. Joint Conf. on AI*, 2001.

[19] S. Wasserman and K. Faust. *Social network analysis: methods and applications*. Cambridge University Press, Cambridge, 1994.