Under consideration for publication in Knowledge and Information Systems

Collective Mining of Bayesian Networks from Distributed Heterogeneous Data

R. Chen[1], K. Sivakumar[1], and H. Kargupta[2]

[1] School of Electrical Engineering and Computer Science,
Washington State University, Pullman, WA 99163, USA;
[2] Department of Computer Science and Electrical Engineering,
University of Maryland Baltimore County, Baltimore, MD 21250, USA

Abstract. We present a collective approach to learning a Bayesian network from distributed heterogenous data. In this approach, we first learn a local Bayesian network at each site using the local data. Then each site identifies the observations that are most likely to be evidence of coupling between local and non-local variables and transmits a subset of these observations to a central site. Another Bayesian network is learnt at the central site using the data transmitted from the local site. The local and central Bayesian networks are combined to obtain a collective Bayesian network, that models the entire data. Experimental results and theoretical justification that demonstrate the feasibility of our approach are presented.

Keywords: Bayesian Network, Web Log Mining, Collective Data Mining, Distributed Data Mining, Heterogenous Data

1. Introduction

Raw data is useful only when it is transformed into knowledge or useful information. This involves data analysis and transformation to extract interesting patterns and correlations among the problem variables. In practical applications, such transformations require efficient data access, analysis, and presentation of the outcome in a timely manner. For example, web server log contains records of user interactions when request for the resources in the servers is received. This contains a wealth of data for the analysis of web usage and identifying different patterns. The advent of large distributed environments in both scientific and

Received xxx
Revised xxx
Accepted xxx

Table 1. Homogeneous case: Site A with a table for credit card transaction records.

Account Number	Amount	Location	Previous record	Unusual transaction
11992346	-42.84	Seattle	Poor	Yes
12993339	2613.33	Seattle	Good	No
45633341	432.42	Portland	Okay	No
55564999	128.32	Spokane	Okay	Yes

Table 2. Homogeneous case: Site B with a table for credit card transaction records.

Account Number	Amount	Location	Previous record	Unusual transaction
87992364	446.32	Berkeley	Good	No
67845921	978.24	Orinda	Good	Yes
85621341	719.42	Walnut	Okay	No
95345998	-256.40	Francisco	Bad	Yes

commercial domains (e.g. the Internet and corporate intranets) introduces a new dimension to this process — a large number of distributed sources of data that can be used for discovering knowledge. Cost of data communication between the distributed databases is a significant factor in an increasingly mobile and connected world with a large number of distributed data sources. This cost consists of several components like (a) Limited network bandwidth, (b) data security, and (c) existing organizational structure of the applications environment. The field of Distributed Knowledge Discovery and Data Mining (DDM) studies algorithms, systems, and human-computer interaction issues for knowledge discovery applications in distributed environments for minimizing this cost.

In this paper, we consider a Bayesian network (BN) model to represent uncertain knowledge. Specifically, we address the problem of learning a BN from heterogenous distributed data. It uses a collective data mining (CDM) approach introduced earlier by Kargupta et. al. [32, 34, 35, 37]. Section 2 provides some background and reviews existing literature in this area. Section 3 presents the collective Bayesian learning technique. Experimental results for two datasets — one simulated and one real world — are presented in Section 4. We would like to mention that these experiments are intended mainly to serve as a proof-of-concept. More extensive results for real data mining applications along the web mining line would be published later (some preliminary results have been presented in [13, 14]). Finally, Section 5 provides some concluding remarks and directions for future work.

2. Background, Motivation, and Related Work

In this section, we provide and background and motivation to the problem by means of an example. We then review the existing literature in this area.

Distributed data mining (DDM) must deal with different possibilities of data distribution. Different sites may contain data for a common set of features of the problem domain. In case of relational data this would mean a consistent database schema across all the sites. This is the homogeneous case. Tables 1 and 2 illustrate

Table 3. Heterogeneous case: Site X with two tables, one for weather and the other for demography.

City	Temp.	Humidity	Wind Chill
Boise	20	24%	10
Spokane	32	48%	12
Seattle	63	88%	4
Portland	51	86%	4
Vancouver	47	52%	6

City	State	Size	Average earning	Proportion of small businesses
Boise	ID	Small	Low	0.041
Spokane	WA	Medium	Medium	0.022
Seattle	WA	Large	High	0.014
Portland	OR	Large	High	0.017
Vancouver	BC	Medium	Medium	0.031

this case using an example from a hypothetical credit card transaction domain.[1] There are two data sites A and B, connected by a network. The DDM-objective in such a domain may be to find patterns of fraudulent transactions. Note that both the tables have the same schema. The underlying distribution of the data may or may not be identical across different data sites.

In the general case the data sites may be *heterogeneous*. In other words, sites may contain tables with different schemata. Different features are observed at different sites. Let us illustrate this case with relational data. Table 3 shows two data-tables at site X. The upper table contains weather-related data and the lower one contains demographic data. Table 4 shows the content of site Y, which contains holiday toy sales data. The objective of the DDM process may be detecting relations between the toy sales, the demographic and weather related features. In the general heterogeneous case the tables may be related through different sets of key indices. For example, Tables 3(upper) and (lower) are related through the key feature *City*; on the other hand Table 3 (lower) and Table 4 are related through key feature *State*. We consider the heterogenous data scenario in this paper.

We would like to mention that heterogenous databases, in general, could be more complicated than the above scenario. For example, there maybe a set of overlapping features that are observed at more than one site. Moreover, the existence of a key that can be used to link together observations across sites is crucial to our approach. For example, for a web log mining application, the key that can be used to link together observations across sites could be produced using either a "cookie" or the user IP address (in combination with other log data like time of access). However, these assumptions are not overly restrictive, and are required for a reasonable solution to the distributed Bayesian learning problem.

[1] Please note that the credit card domain may not always have consistent schema. The domain is used just for illustration.

Table 4. Heterogeneous case: Site Y with one table holiday toy sales.

State	Best Selling Item	Price ($)	Number Items Sold (In thousands)
WA	Snarc Action Figure	47.99	23
ID	Power Toads	23.50	2
BC	Light Saber	19.99	5
OR	Super Squirter	24.99	142
CA	Super Fun Ball	9.99	24

2.1. Motivation

Bayesian networks offer very useful information about the mutual dependencies among the features in the application domain. Such information can be used for gaining better understanding about the dynamics of the process under observation. Financial data analysis, manufacturing process monitoring, sensor data analysis, web mining are a few examples where mining Bayesian networks has been quite useful. Bayesian techniques will also be useful for mining distributed data. In this section we discuss one such scenario and explain how the proposed collective Bayesian learning algorithm can be useful in practice.

Consider a subscriber of a wireless network. This person travels frequently and uses her palmtop computer and cell phone to do business and personal transactions. Her transactions go through different servers depending upon her location during the transaction. Now let us say her wireless service provider wants to offer more personalized service to her by paying careful attention to her needs and tastes. This may be useful for choosing the instant messages appropriate for her taste and needs. For example, if she is visiting the Baltimore area the company may choose to send her instant messages regarding the area Sushi and Italian restaurants that she usually prefers. Since too many of such instant messages are likely to be considered a nuisance, accurate personalization is very important. This is indeed quite well appreciated by the business community and use of Bayesian techniques for personalizing web sites has already been reported elsewhere [51, 52, 5, 6].

The scenario described here is however somewhat different from the traditional web personalization applications where web-log data are centrally located. In the current case the transaction data are logged at different locations since the user is mobile and the wireless service provider is interested in analyzing the overall transactional patterns of the user. As the user moves from one wireless cell to another the servers change and therefore the transactions go through different servers. Today the major wireless service providers support millions of customers. Centralizing this inherently distributed large volume of data may not be scalable. Moreover, these transaction data are heterogeneous. There is no guarantee that the user will perform only a certain type of transactions at every location. The user may choose to perform a wide variety of transactions (e.g. purchasing gifts, money transaction, monitoring the stock portfolio, reading news, and ordering pizza) at different sites. Therefore the features defining the transactions observed at different sites are likely to be different in general although we may have some overlapping characteristic features (e.g. monitoring the stock portfolio everyday!).

As pointed out elsewhere [38] there are many similar situations where data

are distributed among a large number of sites and centralized data mining is not scalable. The technique proposed here will be applicable to many such domains.

2.2. Related Work

The volume of DDM literature is growing fast. There exist a reasonably large body of work on DDM architectures and data mining techniques for the homogeneous and heterogeneous cases. In the following, we review only the existing literature for heterogeneous DDM.

Mining from heterogeneous data constitutes an important class of DDM problems. This issue is discussed in [56] from the perspective of inductive bias. The WoRLD system [2] addressed the problem of concept learning from heterogeneous sites by developing an "activation spreading" approach that is based on first order statistical estimation of the underlying distribution. A novel approach to learn association rules from heterogeneous tables is proposed in [19]. This approach exploits the foreign key relationships for the case of a star schema to develop decentralized algorithms that execute concurrently on the separate tables, and subsequently merge the results. An order statistics-based technique for combining high-variance models generated from heterogeneous sites is proposed in [66].

Kargupta and his colleagues [37] also considered the heterogenous case and proposed the *Collective* framework to address data analysis for heterogeneous environments. They proposed the *Collective Data Mining* (CDM) framework for predictive data modeling that makes use of orthonormal basis functions for correct local analysis. They proposed a technique for distributed decision tree construction [37] and wavelet-based multi-variate regression [32]. Several distributed clustering techniques based on the Collective framework are proposed elsewhere [34, 36]. They also proposed the collective PCA technique [36, 35] and its extension to a distributed clustering application. Additional work on distributed decision tree learning [4], clustering [47, 50, 57], genetic learning[49] DDM design optimization [67], classifier pruning [55], DDM architecture [40], and problem decomposition and local model selection in DDM [45], are also reported.

We now review important literature on learning using Bayesian networks (BN). A BN is a probabilistic graphical model that represents uncertain knowledge [53, 33, 11]. Learning parameters of a Bayesian network from complete data is discussed in [60, 10]. Learning parameters from incomplete data using gradient methods is discussed in [7, 63]. Lauritzen [43] has proposed an EM algorithm to learn Bayesian network parameters, whereas Bauer et. al. [3] describe methods for accelerating convergence of the EM algorithm. Learning using Gibbs sampling is proposed in [65, 27]. The Bayesian score to learn the structure of a Bayesian network is discussed in [18, 10, 29]. Learning the structure of a Bayesian network based on the Minimal Description Length (MDL) principle is presented in [8, 41, 62]. Learning BN structure using greedy hill-climbing and other variants is introduced in [30], whereas Chickering [16] introduced a method based on search over equivalence network classes. Methods for approximating full Bayesian model averaging are presented in [10, 30, 46].

Learning the structure of Bayesian network from incomplete data, is considered in [17, 12, 22, 23, 48, 58, 64]. The relationship between causality and Bayesian networks is discussed in [30, 54, 61, 31]. See [10, 25, 41] for discussion on how to sequentially update the structure of a network as more data is

available. Applications of Bayesian network to clustering (AutoClass) and classification is discussed in [12, 21, 24, 59]. Zweig and Russel [68] use Bayesian networks for speech recognition, whereas Breese et. al. [9] discuss collaborative filtering methods that use Bayesian network learning algorithms. Applications to causal learning in social sciences is discussed in [61]. In [42] the authors report a technique to automatically produce a Bayesian belief network from discovered knowledge using a distributed approach.

An important problem is how to learn the Bayesian network from data in distributed sites. The centralized solution to this problem is to download all datasets from distributed sites. Kenji [39] worked on the homogeneous distributed learning scenario. In this case, every distributed site has the same feature but different observations. In this paper, we address the heterogenous case, where each site has data about only a subset of the features. To our knowledge, there is no significant work that addresses the heterogenous case.

3. Collective Bayesian Learning

In the following, we briefly review Bayesian networks and then discuss our collective approach to learning a Bayesian network that is specifically designed for a distributed data scenario.

3.1. Bayesian Networks: A review

A Bayesian network (BN) is a probabilistic graph model. It can be defined as a pair $(\mathcal{G}, p)$, where $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a directed acyclic graph (DAG) [33, 28]. Here, $\mathcal{V}$ is the vertex set which represents variables in the problem and $\mathcal{E}$ is the edge set which denotes probabilistic relationships among the variables. For a variable $X \in \mathcal{V}$, a parent of X is a node from which there is a directed link to X. Let $pa(X)$ denote the set of parents of X, then the conditional independence property can be represented as follows:

$$P(X \mid \mathcal{V} \setminus X) = P(X \mid pa(X)). \tag{1}$$

This property can simplify the computations in a Bayesian network model. For example, the joint distribution of the set of all variables in $\mathcal{V}$ can be written as a product of conditional probabilities as follows:

$$P(\mathcal{V}) = \prod_{X \in \mathcal{V}} P(X \mid pa(X)). \tag{2}$$

The conditional independence between variables is either obtained from a priori expert knowledge or discerned from data, or a combination of both [33]. The set of conditional distributions $\{P(X \mid pa(X)), X \in \mathcal{V}\}$ are called the parameters of a Bayesian network. Note that if variable X has no parents, then $P(X \mid pa(X)) = P(X)$ is the marginal distribution of X.

Figure 1 is a Bayesian network called the ASIA model (adapted from [44]). The variables are Dyspnoea, Tuberculosis, Lung cancer, Bronchitis, Asia, X-ray, Either, and Smoking. They are all binary variables. The joint distribution of all variables is

$$\begin{aligned} P(A, S, T, L, B, E, X, D) = {} & P(A)P(S)P(T \mid A)P(L \mid S)P(B \mid S) \\ & P(E \mid T, L)P(X \mid E)P(D \mid B, E). \end{aligned} \tag{3}$$

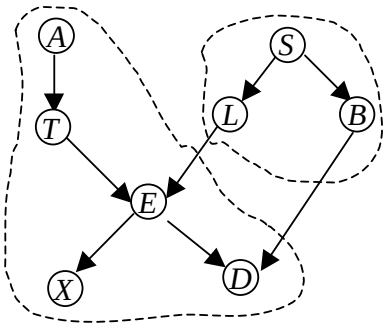

Fig. 1. ASIA Model

The ordering of variables constitutes a constraint on the structure of a Bayesian network. If variable X appears before variable Y, then Y can not be a parent of X. We use the ordering (A, S, T, L, B, E, X, D) as prior knowledge in our example.

Two important issues in using a Bayesian network are : (a) learning a Bayesian network and (b) probabilistic inference. Learning a BN involves learning the structure of the network (the directed graph), and obtaining the conditional probabilities (parameters) associated with the network. Once a Bayesian network is constructed, we usually need to determine various probabilities of interest from the model. This process is referred to as probabilistic inference. For example, in the ASIA model, a diagnosis application would require finding the probability $P(B \mid D)$ of Bronchitis, given the (observed) symptom Dyspnoea. This probability (usually called posterior probability) can be computed using the Bayes rule.

3.2. Collective Bayesian Network Learning Strategy

We now present a collective strategy to learn a Bayesian network (both structure and parameters) when data is distributed among different sites. The centralized solution to this problem is to download all datasets from distributed sites to a central site. In many applications, this would not be feasible because of the size of the data, available communication bandwidth, or due to security considerations. Learning a BN for the homogeneous case was studied by Kenji [39]. In this case, every distributed site has the same set of features but has different set of observations. We address here the heterogenous case, where each distributed site has all the observations for only a subset of the features.

The primary steps in our approach are:

- Compute local BNs (local model) involving the variables observed at each site (local variables) based on local data.
- At each site, based on the local BN, identify the observations that are most likely to be evidence of coupling between local and non-local variables. Transmit a subset of these observations to a central site.
- At the central site, a limited number of observations of all the variables are now available. Using this, compute a non-local BN consisting of links between variables across two or more sites.

– Combine the local models with the links discovered at the central site to obtain a collective BN.

The non-local BN thus constructed would be effective in identifying associations between variables across sites, whereas the local BNs would detect associations among local variables at each site. The conditional probabilities can also be estimated in a similar manner. Those probabilities that involve only variables from a single site can be estimated locally, whereas the ones that involve variables from different sites can be estimated at the central site. Same methodology could be used to update the network based on new data. First, the new data is tested for how well it fits with the local model. If there is an acceptable statistical fit, the observation is used to update the local conditional probability estimates. Otherwise, it is also transmitted to the central site to update the appropriate conditional probabilities (of cross terms). Finally, a collective BN can be obtained by taking the union of nodes and edges of the local BNs and the nonlocal BN and using the conditional probabilities from the appropriate BNs. Probabilistic inference can now be performed based on this collective BN. Note that transmitting the local BNs to the central site would involve a significantly lower communication as compared to transmitting the local data.

It is quite evident that learning probabilistic relationships between variables that belong to a single local site is straightforward and does not pose any additional difficulty as compared to a centralized approach.[2] The important objective is to correctly identify the coupling between variables that belong to two (or more) sites. These correspond to the edges in the graph that connect variables between two sites and the conditional probability(ies) at the associated node(s). In the following, we describe our approach to selecting observations at the local sites that are most likely to be evidence of strong coupling between variables at two different sites.

3.3. Selection of samples for transmission to global site

For simplicity, we will assume that the data is distributed between two sites and will illustrate the approach using the BN in Figure 1. The extension of this approach to more than two sites is straightforward. Let us denote by $\mathcal{A}$ and $\mathcal{B}$, the variables in the left and right groups, respectively, in Figure 1. We assume that the observations for $\mathcal{A}$ are available at site A, whereas the observations for $\mathcal{B}$ are available at a different site B. Furthermore, we assume that there is a common feature ("key" or index) that can be used to associate a given observation in site A to a corresponding observation in site B. Naturally, $\mathcal{V} = \mathcal{A} \cup \mathcal{B}$.

At each local site, a local Bayesian network can be learned using only samples in this site. This would give a BN structure involving only the local variables at each site and the associated conditional probabilities. Let $p_A(.)$ and $p_B(.)$ denote the estimated probability function involving the local variables. This is the product of the conditional probabilities as indicated by (2). Since $p_A(x)$, $p_B(x)$ denote the probability or likelihood of obtaining observation x at sites A and B, we would call these probability functions the likelihood functions $l_A(.)$ and $l_B(.)$, for the local model obtained at sites A and B, respectively. The observations

[2] This may not be true for arbitrary Bayesian network structure. We will discuss this issue further in the last section.

at each site are ranked based on how well it fits the local model, using the local likelihood functions. The observations at site A with large likelihood under $l_A(.)$ are evidence of "local relationships" between site A variables, whereas those with low likelihoods under $l_A(.)$ are possible evidence of "cross relationships" between variables across sites. Let $S(A)$ denote the set of keys associated with the latter observations (those with low likelihood under $l_A(.)$). In practice, this step can be implemented in different ways. For example, we can set a threshold ρ_A and if $l_A(x) \leq \rho_A$, then $x \in S_A$. The sites A and B transmit the set of keys S_A, S_B, respectively, to a central site, where the intersection $S = S_A \cap S_B$ is computed. The observations corresponding to the set of keys in S are then obtained from each of the local sites by the central site.

The following argument justifies our selection strategy. Using the rules of probability, and the assumed conditional independence in the BN of Figure 1, it is easy to show that:

$$P(\mathcal{V}) = P(\mathcal{A}, \mathcal{B}) = P(\mathcal{A} \mid \mathcal{B})P(\mathcal{B}) = P(\mathcal{A} \mid nb(\mathcal{A}))P(\mathcal{B}), \tag{4}$$

where $nb(\mathcal{A}) = \{B, L\}$ is the set of variables in $\mathcal{B}$, which have a link connecting it to a variable in $\mathcal{A}$. In particular,

$$P(\mathcal{A} \mid nb(\mathcal{A})) = P(A)P(T \mid A)P(X \mid E)P(E \mid T, L)P(D \mid E, B). \tag{5}$$

Note that, the first three terms in the right-hand side of (5) involve variables local to site A, whereas the last two terms are the so-called *cross terms*, involving variables from sites A and B. Similarly, it can be shown that

$$P(\mathcal{V}) = P(\mathcal{A}, \mathcal{B}) = P(\mathcal{B} \mid \mathcal{A})P(\mathcal{A}) = P(\mathcal{B} \mid nb(\mathcal{B}))P(\mathcal{A}), \tag{6}$$

where $nb(\mathcal{B}) = \{E, D\}$ and

$$P(\mathcal{B} \mid nb(\mathcal{B})) = P(S)P(B \mid S)P(L \mid S)P(E \mid T, L)P(D \mid E, B). \tag{7}$$

Therefore, an observation $\{A = a, T = t, E = e, X = x, D = d, S = s, L = l, B = b\}$ with low likelihood at both sites A and B; i.e. for which both $P(\mathcal{A})$ and $P(\mathcal{B})$ are small, is an indication that both $P(\mathcal{A} \mid nb(\mathcal{A}))$ and $P(\mathcal{B} \mid nb(\mathcal{B}))$ are large for that observation (since observations with small $P(\mathcal{V})$ are less likely to occur). Notice from (5) and (7) that the terms common to both $P(\mathcal{A} \mid nb(\mathcal{A}))$ and $P(\mathcal{B} \mid nb(\mathcal{B}))$ are precisely the conditional probabilities that involve variables from both sites A and B. In other words, this is an observation that indicates a coupling of variables between sites A and B and should hence be transmitted to a central site to identify the specific coupling links and the associated conditional probabilities.

In a sense, our approach to learning the cross terms in the BN involves a selective sampling of the given dataset that is most relevant to the identification of coupling between the sites. This is a type of *importance sampling*, where we select the observations that have high conditional probabilities corresponding to the terms involving variables from both sites. Naturally, when the values of the different variables (features) from the different sites, corresponding to these selected observations are pooled together at the central site, we can learn the coupling links as well as estimate the associated conditional distributions. These selected observations will, by design, not be useful to identify the links in the BN that are local to the individual sites. This has been verified in our experiments (see Section 4).

3.4. Performance Analysis

In the following, we present a brief theoretical analysis of the performance of the proposed collective learning method. We compare the performance of our collective BN with that of a Bayesian network learned using a centralized approach (referred to as centralized BN in the sequel).

There are two types of errors involved in learning a BN: (a) Error in BN structure and (b) Error in parameters (probabilities) of the BN. The structure error is defined as the sum of the number of correct edges missed and the number of incorrect edges detected. For parameter error, we need to quantify the "distance" between two probability distributions. We only consider learning error in the parameters, assuming that the structure of the BN has been correctly determined (or is given). A widely used metric is the Kullback-Leibler (KL) distance (cross-entropy measure) $d_{KL}(p, q)$ between two discrete probabilities, $\{p_i\}$, $\{q_i\}$, $i = 1, 2, \ldots, N$

$$d_{KL}(p, q) = \sum_{i=1}^{N} p_i \ln(\frac{p_i}{q_i}) \tag{8}$$

where N is the number of possible outcomes.

Indeed, if p^* is the empirically observed distribution for data samples $\{s_i, 1 \leq i \leq M\}$ and h is a hypothesis (candidate probability distribution for the underlying true distribution), then [1]

$$\begin{aligned} d_{KL}(p^*, h) &= \sum_{i=1}^{M} p^*(s_i) \ln(\frac{p^*(s_i)}{h(s_i)}) = \sum_{i=1}^{M} \frac{1}{M} \ln \frac{1}{M} - \sum_{i=1}^{M} \frac{1}{M} \ln(h(s_i)) \\ &= \ln \frac{1}{M} - \frac{1}{M} \sum_{i=1}^{M} \ln(h(s_i)). \end{aligned} \tag{9}$$

Therefore, minimizing the KL distance with respect to the empirically observed distribution is equivalent to finding the maximum likelihood solution h^* of $\sum_{i=1}^{M} \ln(h(s_i))$.

Since the BN provides a natural factorization of the joint probability in terms of the conditional probabilities at each node (see (2)), it is convenient to express the KL distance between two joint distributions in terms of the corresponding conditional distributions. Let h and c be two possible (joint) distributions of the variables in a BN. For $i = 1, 2, \ldots, n$, let $h_i(x_i \mid \pi_i)$, $c_i(x_i \mid \pi_i)$ be the corresponding conditional distribution at node i, where x_i is the variable at node i and π_i is the set of parents of node i. Following [20], define a distance $d_{CP}(P, c_i, h_i)$ between h_i and c_i with respect to the true distribution P:

$$d_{CP}(P, c_i, h_i) = \sum_{\pi_i} P(\pi_i) \sum_{x_i} P(x_i \mid \pi_i) \ln(\frac{c_i(x_i \mid \pi_i)}{h_i(x_i \mid \pi_i)}). \tag{10}$$

It is then easy to show that

$$d_{KL}(P, h) - d_{KL}(P, c) = \sum_{i=1}^{n} d_{CP}(P, c_i, h_i). \tag{11}$$

Equations (10) and (11) provide a useful decomposition of the KL distance be-

tween the true distribution P and two different hypotheses c, h. This will be useful in our analysis of sample complexity in the following sub-section.

3.5. Sample Complexity

We now derive a relationship between the accuracy of collective BN and the number of samples transmitted to the central site. We consider the unrestricted multinomial class BN, where all the node variables are Boolean. The hypothesis class H is determined by the set of possible conditional distributions for the different nodes. Given a BN of n variables and a hypothesis class H, we need to choose a hypothesis $h \in H$ which is close to a unknown distribution P. Given an error threshold ϵ and a confidence threshold δ, we are interested in constructing a function $N(\epsilon, \delta)$, such that if the number of samples M is larger than $N(\epsilon, \delta)$

$$Prob(d_{KL}(P, h) < d_{KL}(P, h_{opt}) + \epsilon) > 1 - \delta, \tag{12}$$

where $h_{opt} \in H$ is the hypothesis that minimizes $d_{KL}(P, h)$. If smallest value of $N(\epsilon, \delta)$ that satisfies this requirement is called the sample complexity. This is usually referred to as the probably approximately correct (PAC) framework. Friedman and Yakhini [26] have examined the sample complexity of the maximum description length principle (MDL) based learning procedure for BNs.

Dasgupta [20] gave a thorough analysis for the multinomial model with Boolean variables. Suppose the BN has n nodes and each node has at most k parents. Given ϵ and δ, an upper bound of sample complexity is

$$N(\epsilon, \delta) = \frac{288 n^2 2^k}{\epsilon^2} \ln^2(1 + \frac{3n}{\epsilon} \ln \frac{18 n^2 2^k \ln(1 + 3n/\epsilon)}{\epsilon \delta}). \tag{13}$$

Equation (13) gives a relation between the sample size and the (ϵ, δ) bound. For the conditional probability $h_i(x_i \mid \pi_i) = P(X_i = x_i \mid \Pi_i = \pi_i)$, we have (see (10))

$$d_{CP}(P, h_{opt}, h) \leq \frac{\epsilon}{n} \tag{14}$$

We now use the above ideas to compare the performance of the collective learning method with the centralized method. We fix the confidence δ and suppose that an ϵ^{cen} can be found for the centralized method, for a given sample size M using (13). Then, following the analysis in [20, Section 5],

$$d_{CP}(P, h_{opt}^{cen}, h^{cen}) \leq \frac{\epsilon^{cen}}{n}, \tag{15}$$

where h_{opt}^{cen} is the optimal hypothesis and h^{cen} is the hypothesis obtained based on a centralized approach. Then from (11)

$$d_{KL}(P, h^{cen}) - d_{KL}(P, h_{opt}^{cen}) = \sum_{i=1}^{n} d_{CP}(P, h_{i,opt}^{cen}, h_i^{cen}) \leq \sum_{i=1}^{n} \frac{\epsilon^{cen}}{n} = \epsilon^{cen}. \tag{16}$$

For the collective BN learning method, the set of nodes can be split into two parts. Let V_l be the set of nodes, which have all their parent nodes at the same local site, and V_c be the set of nodes, which have at least one parent node belonging to a site different than the node itself. For ASIA model, $V_l = \{A, S, T, L, B, X\}$ and $V_c = \{E, D\}$. We use n_l and n_c to denote the cardinality

of the sets V_l and V_c. If a node $x \in V_l$, the collective method can learn the conditional probability $P(x \mid pa(x))$ using all data because this depends only on the local variables. Therefore, for $x \in V_l$,

$$d_{CP}(P, h^{col}_{opt}, h^{col}) \leq \frac{\epsilon^{col}_1}{n} = \frac{\epsilon^{cen}}{n}, \tag{17}$$

where, for the local terms, $\epsilon^{col}_1 = \epsilon^{cen}$. For the nodes in V_c, only the data transmitted to the central site can be used to learn its conditional probability. Suppose M_c data samples are transmitted to the central site, and the error threshold ϵ^{col}_2 satisfies (13), for the same fixed confidence $1 - \delta$. Therefore, for $x \in V_c$, we have from (14) that $d_{CP}(P, h^{col}_{opt}, h^{col}) \leq \frac{\epsilon^{col}_2}{n}$, where $\epsilon^{col}_2 \geq \epsilon^{cen}$, in general, since the in the collective learning method, only $M_c \leq M$ samples are available at the central site. Then from (11) and (17)

$$\begin{aligned} d_{KL}(P, h^{col}_{opt}) - d_{KL}(P, h^{col}) &= \sum_{i=1}^{n} d_{CP}(P, h^{col}_{i,opt}, h^{col}_i) \\ &= \sum_{i \in V_l} d_{CP}(P, h^{col}_{i,opt}, h^{col}_i) + \sum_{i \in V_c} d_{CP}(P, h^{col}_{i,opt}, h^{col}_i) \\ &= \frac{n_l}{n}\epsilon^{cen} + \frac{n_c}{n}\epsilon^{col}_2 \end{aligned} \tag{18}$$

Comparing (16) and (18), it is easy to see that the error threshold of the collective method is $\epsilon^{col} = \frac{n_l}{n}\epsilon^{cen} + \frac{n_c}{n}\epsilon^{col}_2$. The difference of the error threshold between the collective and the centralized method is

$$\epsilon^{col} - \epsilon^{cen} = \frac{n_c}{n}(\epsilon^{col}_2 - \epsilon^{cen}) \tag{19}$$

Equation (19) shows two important properties of the collective method. First, the difference in performance is independent of the variables in V_l. This means the performance of the collective method for the parameters of local variables is same as that of the centralized method. Second, the collective method is a tradeoff between accuracy and the communication overhead. The more data we communicate, more closely ϵ^{col}_2 will be to ϵ^{cen}. When $M_c = M$, $\epsilon^{col}_2 = \epsilon^{cen}$, and $\epsilon^{col} - \epsilon^{cen} = 0$.

4. Experimental Results

We tested our approach on three different datasets — ASIA model, real web log data, and simulated web log data. We present our results for the three cases in the following subsections.

4.1. ASIA Model

This experiment illustrates the ability of the proposed collective learning approach to correctly obtain the structure of the BN (including the cross-links) as well as the parameters of the BN. Our experiments were performed on a dataset that was generated from the BN depicted in Figure 1 (ASIA Model).

No.	T	L	E	Probability
1	F	F	F	0.9
2	T	F	F	0.1
3	F	T	F	0.1
4	T	T	F	0.01
5	F	F	T	0.1
6	T	F	T	0.9
7	F	T	T	0.9
8	T	T	T	0.99

A	0.99	0.01						
S	0.5	0.5						
T	0.1	0.9	0.9	0.1				
L	0.3	0.6	0.7	0.4				
B	0.1	0.8	0.9	0.2				
E	0.9	0.1	0.1	0.01	0.1	0.9	0.9	0.99
X	0.2	0.6	0.8	0.4				
D	0.9	0.1	0.1	0.01	0.1	0.9	0.9	0.99

Table 5. (Top) The conditional probability of node E and (Bottom) All conditional probabilities for the ASIA model

The conditional probability of a variable is a multidimensional array, where the dimensions are arranged in the same order as ordering of the variables, viz. $\{A, S, T, L, B, E, X, D\}$. Table 5 (top) depicts the conditional probability of node E. It is laid out such that the first dimension toggles fastest. From Table 5, we can write the conditional probability of node E as a single vector as follows: $[0.9, 0.1, 0.1, 0.01, 0.1, 0.9, 0.9, 0.99]$. The conditional probabilities (parameters) of ASIA model are given in Table 5 (bottom) following this ordering scheme. We generated $n = 6000$ observations from this model, which were split into two sites as illustrated in Figure 1 (site A with variables A, T, E, X, D and site B with variables S, L, B). Note that there are two edges ($L \rightarrow E$ and $B \rightarrow D$) that connect variables from site A to site B, the rest of the six edges being local.

Local Bayesian networks were constructed using a conditional independence test based algorithm [15], for learning the BN structure and a maximum likelihood based method for estimating the conditional probabilities. The local networks were exact as far as the edges involving only the local variables. We then tested the ability of the collective approach to detect the two non-local edges. The estimated parameters of these two local Bayesian network is depicted in Table 6. Clearly, the estimated probabilities at all nodes, except nodes E and D, are close to the true probabilities given in Table 5. In other words, the parameters that involve only local variables have been successfully learnt at the local sites.

A fraction of the samples, whose likelihood are smaller than a selected threshold T, were identified at each site. In our experiments, we set

$$T_i = \mu_i + \alpha\sigma_i, \quad i \in \{A, B\}, \tag{20}$$

for some constant α, where μ_i is the (empirical) mean of the local likelihood values and σ_i is the (empirical) standard deviation of the local likelihood values. The samples with likelihood less than the threshold (T_A at site A T_B at site B)

Local A				
A	0.99	0.01		
T	0.10	0.84	0.90	0.16
E	0.50	0.05	0.50	0.95
X	0.20	0.60	0.80	0.40
D	0.55	0.05	0.45	0.95
Local B				
S	0.49	0.51		
L	0.30	0.59	0.70	0.41
B	0.10	0.81	0.90	0.19

Table 6. The conditional probabilities of local site A and local site B

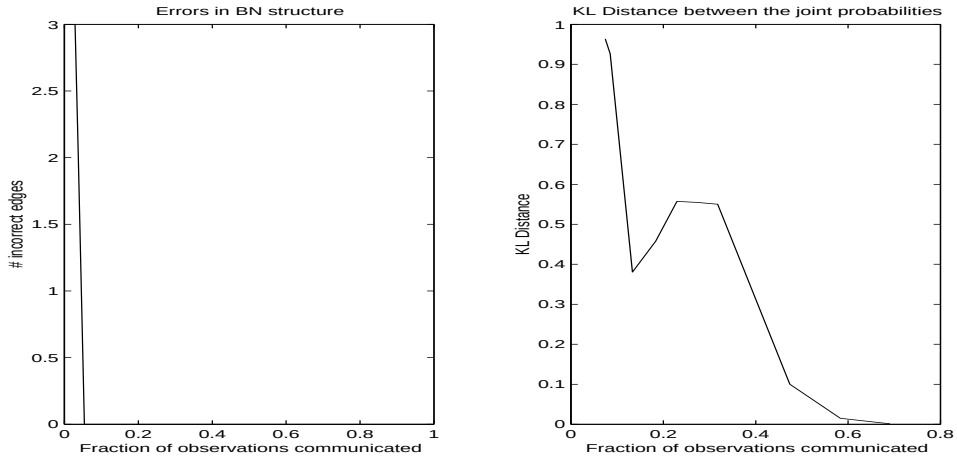

Fig. 2. Performance of collective BN: (left) structure learning error (right) parameter learning error.

at both sites were sent to a central site. The central learns a global BN based on these samples. Finally, a collective BN is formed by taking the union of edges detected locally and those detected at the central site. The error in structure learning of the collective Bayesian network is defined as the sum of the number of correct edges missed and the number of incorrect edges detected. This is done for different values of α. Figure 2 (left) depicts this error as a function of the number of samples communicated (which is determined by α). It is clear that the exact structure can be obtained by transmitting about 5% of the total samples.

Next we assessed the accuracy of the estimated conditional probabilities. For the collective BN, we used the conditional probabilities from local BN for the local terms and the ones estimated at the global site for the cross terms. This was compared with the performance of a BN learnt using a centralized approach (by aggregating all data at a single site). Figure 2 (right) depicts the KL distance $d(p_{cntr}(\mathcal{V}), p_{coll}(\mathcal{V}))$ between the joint probabilities computed using our collective approach and the one computed using a centralized approach. Clearly, even with a small communication overhead, the estimated conditional

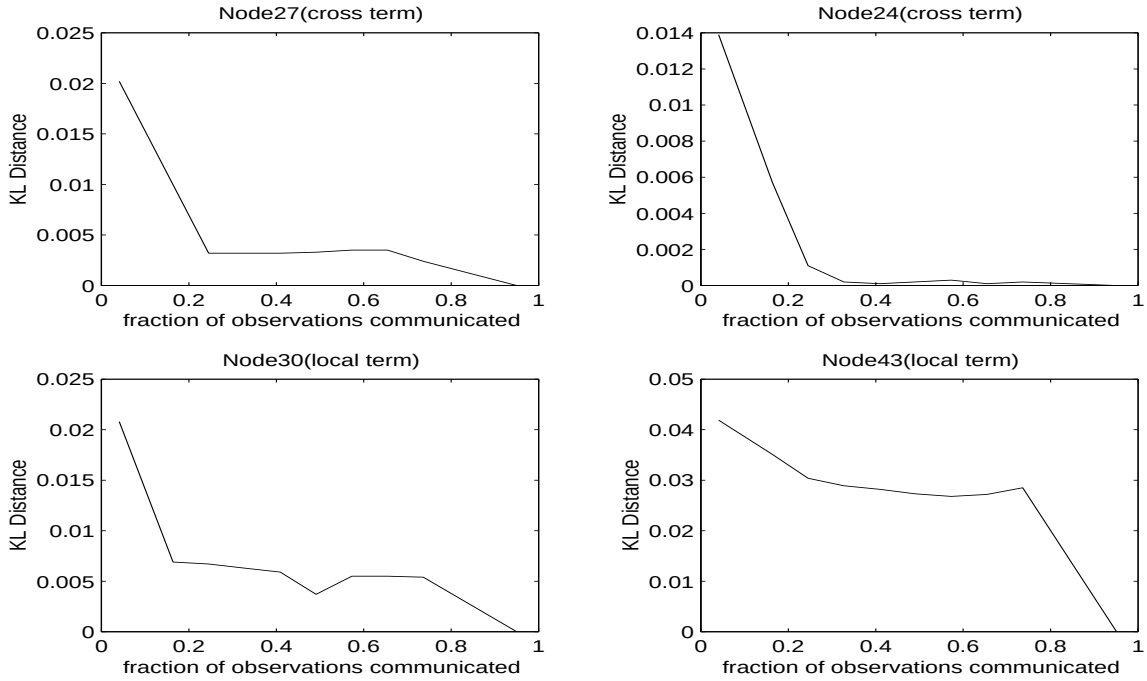

Fig. 3. KL distance between conditional probabilities

probabilities based on our collective approach is quite close that obtained from a centralized approach.

A more important test of our approach is the error in estimating the conditional probabilities at nodes E and D, since these are the cross terms, estimated at the global site, based on a selective transmission of data. The KL distance between the conditional probabilities was computed based on our collective BN and a BN obtained using a centralized approach (by transmitting all data to one site), for the cross terms: $p(E \mid T, L)$ and $p(D \mid E, B)$. Given that these are conditional probabilities, we compute the sum over all the possible values of $\{T, L\}$, of the KL distance between $p_{\text{coll}}(E \mid T, L)$ and $p_{\text{cntr}}(E \mid T, L)$, estimated using our collective approach and the centralized approach, respectively. Figure 3 (top left) depicts the KL distance $\sum_{T,L} d(p_{\text{cntr}}(E \mid T, L), p_{\text{coll}}(E \mid T, L))$, between the two estimates. Figure 3 (top right) depicts the sum $\sum_{B,E} d(p_{\text{cntr}}(D \mid B, E), p_{\text{coll}}(D \mid B, E))$, over all the possible values of $\{E, B\}$, of the KL distance between the two estimates. Clearly, even with a small data communication, the estimates of the conditional probabilities of the cross-terms, based on our collective approach, is quite close to that obtained by the centralized approach. To further verify the validity of our approach, the transmitted data at the central site was used to estimate two local conditional probabilities, $p(X \mid E)$ and $p(L \mid S)$. The corresponding KL distances are depicted in the bottom row of Figure 3 (left: node L and right: node: X). It is clear that the estimates of these probabilities is quite poor, unless a substantial fraction of the data is transmitted. This clearly demonstrates that our technique can be used to perform a biased sampling for discovering relationships between variables across sites.

4.2. Webserver Log Data

In the second set of experiments, we used data from real world domain — a web server log data. This experiment illustrates the ability of the proposed collective learning approach to learn the parameters of a BN from real world web log data. Web server log contains records of user interactions when request for the resources in the servers is received. Web log mining can provide useful information about different user profiles. This in turn can be used to offer personalized services as well as to better design and organize the web resources based on usage history.

In our application, the raw web log file was obtained from the web server of the School of EECS at Washington State University — `http://www.eecs.wsu.edu`. There are three steps in our processing. First we preprocess the raw web log file to transform it to a session form which is useful to our application. Each session corresponds to the logs from a single user in a single web session. We consider each session as a data sample. Then we categorize the resource (html, video, audio etc.) requested from the server into eight categories: E-EE Faculty, C-CS Faculty, L-Lab and facilities, T-Contact Information, A-Admission Information, U-Course Information, H-EECS Home, and R-Research. These categories are our features. Each feature value in a session is set to one or zero, depending on whether the user requested resources corresponding to that category. An 8-feature, binary dataset was thus obtained, which was used to learn a BN. Figure 4 illustrates this process schematically.

A central BN was first obtained using the whole dataset. Figure 5 depicts the structure of this centralized BN. We then split the features into two sets, corresponding to a scenario where the resources are split into two different web servers. Site A has features E, C, T, and U and site B has features L, A, H, and R. We assumed that the BN structure was known, and estimated the parameters (probability distribution) of the BN using our collective BN learning approach. Figure 6 shows the KL distance between the central BN and the collective BN as a function of the fraction of observations communicated. Clearly the parameters of collective BN is close to that of central BN even with a small fraction of data communication.

4.3. Simulated log data

This experiment illustrates the scalability of our approach with respect to number of sites, features, and observations. To this end, we generated a large dataset to simulate web log data. We assume that the users in a wireless network can be divided into several groups, each group having a distinct usage pattern. This can be described by means of the (conditional) probability of a user requesting resource i, given that she has requested resource j. A BN can be used to model such usage patterns. In our simulation, we used 43 features (nodes in the BN) and generated 10000 log samples. The structure of the BN is shown in Figure 7. These 43 features were split into four different sites as follows — Site 1: {1, 5, 10, 15, 16, 22, 23, 24, 30, 31, 37, 38}, Site 2: {2, 6, 7, 11, 17, 18, 25, 26, 32, 39, 40}, Site 3: {3, 8, 12, 19, 20, 27, 33, 34, 41, 42}, Site 4: {4, 9, 13, 14, 21, 28, 29, 35, 36, 42, 43}. Note that there are eight cross edges: $Node6 \rightarrow Node10$, $Node3 \rightarrow Node7$, $Node9 \rightarrow Node12$, $Node17 \rightarrow Node24$, $Node18 \rightarrow Node27$, $Node20 \rightarrow Node28$, $Node33 \rightarrow Node40$, and $Node34 \rightarrow Node42$.

Collect

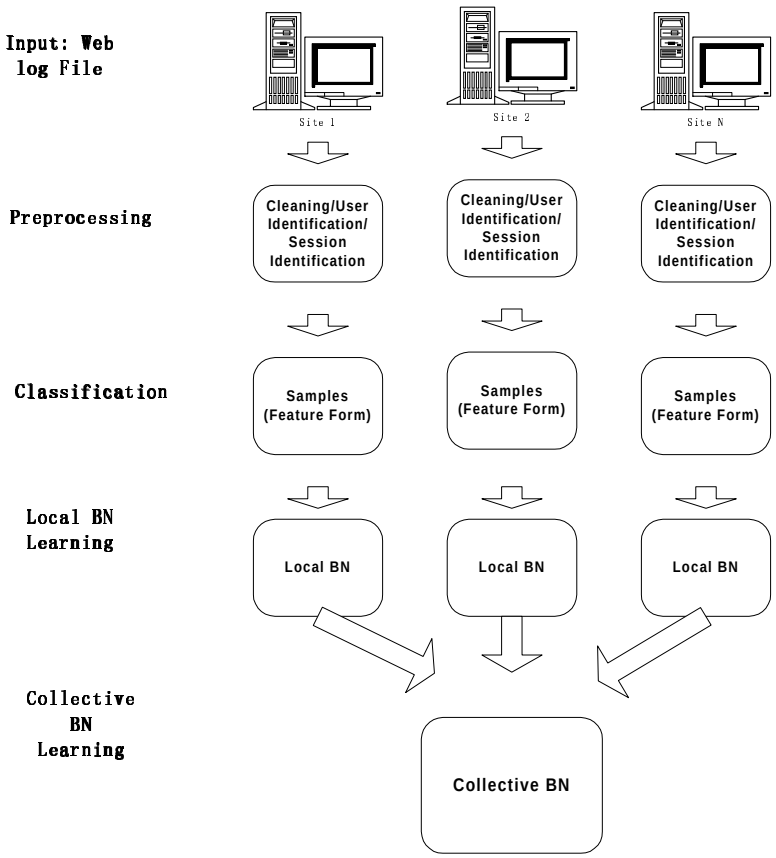

Fig. 4. Schematic illustrating preprocessing and mining of web log data

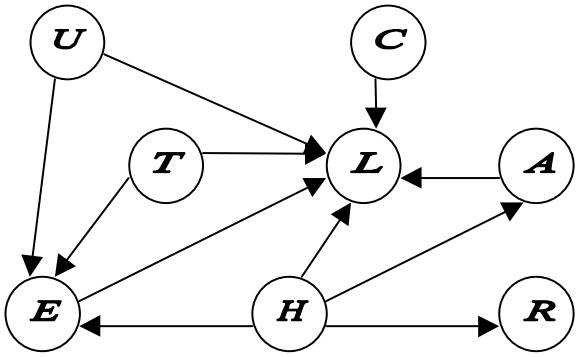

Fig. 5. KL distance between conditional probabilities

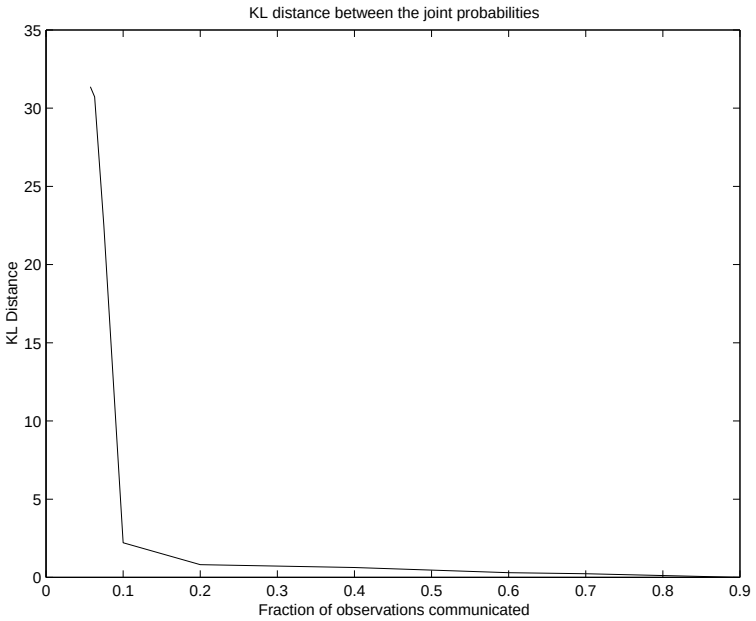

Fig. 6. KL distance between conditional probabilities

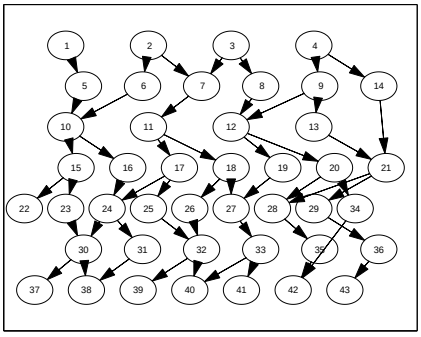

Fig. 7. Structure of BN for web mining simulation

We assumed that the structure of the Bayesian network was given, and tested our approach for estimating the conditional probabilities. The KL distance between the conditional probabilities estimated based on our collective BN and a BN obtained using a centralized approach was computed. In particular, we illustrate the results for the conditional probabilities at four different nodes: 24, 27, 38, and 43; i.e., for $p(Node27 \mid Node18, Node19)$, $p(Node24 \mid Node16, Node17)$, $p(Node38 \mid Node30, Node31)$, and $p(Node43 \mid Node35, Node36)$. Note that the first two conditional probabilities represent cross terms, whereas the last two conditional probabilities represent local terms. Given that these are conditional probabilities, we compute the sum over all the possible values of $\{Node18, Node19\}$, of the KL distance between p_{coll} and p_{cntr}, estimated using our collective approach and the centralized approach, respectively. Figure 8 (top left) depicts the

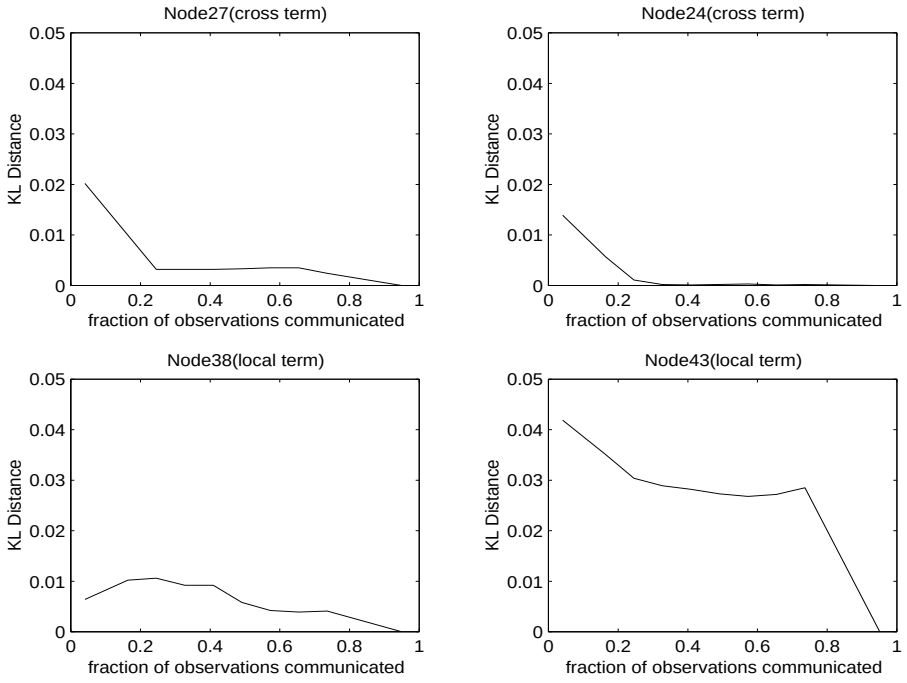

Fig. 8. KL distance between conditional probabilities for simulated web log data experiment

KL distance $\sum_{Node18,Node19} d(p_{\text{cntr}}(Node27 \mid Node18, Node19), p_{\text{coll}}(Node27 \mid Node18, Node19))$, between the two estimates. Figure 8 (top right) depicts a similar KL distance for $Node24$. Clearly, even with a small data communication, the estimates of the conditional probabilities of the cross-terms, based on our collective approach, is quite close to that obtained by the centralized approach.

To further verify the validity of our approach, the transmitted data at the central site was used to estimate two local conditional probabilities of $Node38$ and $Node43$. The corresponding KL distances are depicted in the bottom row of Figure 8 (left: node 38 and right: node 43). It is clear that the estimates of these probabilities is quite poor, unless a substantial fraction of the data is transmitted. Our experiments clearly demonstrates that our technique can be used to perform a biased sampling for discovering relationships between variables across sites. This simulation also illustrates the fact that the proposed approach scales well with respect to number of nodes, samples, and sites.

5. Discussions and Conclusions

We have presented an approach to learning BNs from distributed heterogenous data. This is based on a collective learning strategy, where a local model is obtained at each site and the global associations are determined by a selective transmission of data to a central site. In our experiments, the performance of the collective BN was quite comparable to that obtained from a centralized approach,

even for a small data communication. To our knowledge, this is the first approach to learning BNs from distributed heterogenous data.

Our experiments suggest that the collective learning scales well with respect to number of sites, samples, and features .

Many interesting applications are possible from a BN model of the web log data. For example, specific structures in the overall BN would indicate special user patterns. This could be used to identify new user patterns and accordingly personalize offers and services provided to such users. Another interesting application is to classify the users into different groups based on their usage patterns. This can be thought of decomposing the overall BN (obtained from the log data by collective learning) into a number of sub-BNs, each sub-BN representing a specific group having similar preferences. We are actively pursuing these ideas and would report results in a future publication.

We now discuss some limitations of our proposed approach, which suggest possible directions for future work.

- **Hidden node at local sites**: For certain network structures, it may not be possible to obtain the correct (local) links, based on local data at that site. For example, consider the ASIA model shown in Figure 1, where the observations corresponding to variables A, T, E, and X are available at site A and those corresponding to variables S, L, B, and D are available at site B. In this case, when we learn a local BN at site B, we would expect a (false) edge from node L to node D, because of the edges $L \rightarrow E$ and $E \rightarrow D$ in the overall BN and the fact that node E is "hidden" (unobserved) at site B. This was verified experimentally as well. However, the cross-links $L \rightarrow E$ and $E \rightarrow D$ were still detected correctly at the central site, using our "selctively sampled" data. Therefore, it is necessary to re-examine the local links after discovering the cross-links. In other words, some post-processing of the resulting overall BN is required to eliminate such false local edges. This can be done by evaluating an appropriate score metric on BN configurations with and without such suspect local links. We are currently pursuing this issue. Note, however, that we do not encounter this problem in the examples presented in Section 4.
- **Assumptions about the data**: As mentioned earlier, we assume the existence of a key that links observations across sites. Moreover, we consider a simple heterogenous partition of data, where the variable set at different sites are non-overlapping. We also assume that our data is stationary (all data points come from the same distribution) and free of outliers. These are simplifying assumptions to derive a reasonable algorithm for distributed Bayesian learning. Suitable learning strategies that would allow us to relax of some of these assumptions would be an important area of research.
- **Structure Learning**: Even when the data is centralized, learning the structure of BN is considerably more involved than estimating the parameters or probabilities associated with the network. In a distributed data scenarion, the problem of obtaining the correct network structure is even more pronounced. The "hidden node" problem discussed earlier is one example of this. As in the centralized case, prior domain knowledge at each local site, in the form of probabilistic independence or direct causation, would be very helpful. Our experiments on the ASIA model demonstrate that the proposed collective BN learning approach to obtain the network structure is reasonable, at least for simple cases. However, this is just a beginning and deserves careful investigation.

– **Performance Bounds**: Our approach to "selective sampling" of data that maybe evidence of cross-terms is reasonable based on the discussion in Section 3 (see eq. (4)-(7)). This was verified experimentally for the three examples in Section 4. Currently, we are working towards obtaining bounds for the performance of our collective BN as compared to that obtained from a centralized approach, as a function of the data communication involved.

Acknowledgements. This work was partially supported by NASA, under Cooperative agreement NCC 2-1252.

References

[1] N. Abe, J. Takeuchi, and M. Warmuth, "Polynomial learnability of probabilistic concepts with respect to the Kullback-Leibler divergence," in *Proceedings of the 1991 Workshop on Computational Learning Theory*, pp. 277–289, 1991.

[2] J. Aronis, V. Kulluri, F. Provost, and B. Buchanan, "The WoRLD: Knowledge discovery and multiple distributed databases," in *Proceedings of the Florida Artificial Intellegence Research Symposium (FLAIRS-97)*, pp. 11–14, 1997. Also available as Technical Report ISL-96-6, Intelligent Systems Laborotory, Department of Computer Science, University of Pittsburgh.

[3] E. Bauer, D. Koller, and Y. Singer, "Update rules for parameter estimation in Bayesian networks," in *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence* (D. Geiger and P. Shanoy, eds.), pp. 3–13, Morgan Kaufmann, 1997.

[4] R. Bhatnagar and S. Srinivasan, "Pattern discovery in distributed databases," in *Proceedings of the AAAI-97 Conference*, (Providence), pp. 503–508, AAAI Press, July 1997.

[5] D. Billsus and M. Pazzani, "Revising user profiles: The search for interesting web sites," in *Proceedings of the Third International Workshop on Multistrategy Learning*, AAAI Press, 1996.

[6] D. Billsus and M. Pazzani, "Learning probabilistic user models," in *Workshop notes of Machine Learning for User Modeling — Sixth International Conference on User Modeling*, (Chia Laguna, Sardinia), 1997.

[7] J. Binder, D. Koller, S. Russel, and K. Kanazawa, "Adaptive probabilistic networks with hidden variables," *Machine Learning*, vol. 29, pp. 213–244, 1997.

[8] R. R. Bouckaert, "Properties of Bayesian network learning algorithms," in *Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence* (R. L. de Mantaras and D. Poole, eds.), pp. 102–109, Morgan Kaufmann, 1994.

[9] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborativefiltering," in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence* (G. F. Cooper and S. Moral, eds.), Morgan Kaufmann, 1998.

[10] W. Buntine, "Theory refinement on Bayesian networks," in *Proceedings of the Seventh Annual Conference on Uncertainty in Artificial Intelligence* (B. D. D'Ambrosio and P. S. amd P. P. Bonissone, eds.), pp. 52–60, Morgan Kaufmann, 1991.

[11] E. Charniak, "Bayesian networks without tears," *AI Magazine*, vol. 12, pp. 50–63, 1991.

[12] P. Cheeseman and J. Stutz, "Bayesian classification (autoclass): Theory and results," in *Advances in Knowledge Discovery and Data Mining* (U. Fayyad, G. P. Shapiro, P. Smyth, and R. S. Uthurasamy, eds.), AAAI Press, 1996.

[13] R. Chen, K. Sivakumar, and H. Kargupta, "An approach to online Bayesian learning from multiple data streams," in *Proceedings of the Workshop on Ubiquitous Data Mining: Technology for Mobile and Distributed KDD (In the 5th European Conference, PKDD 2001)* (H. Hargupta, K. Sivakumar, and R. Wirth, eds.), (Freiburg, Germany), pp. 31–45, September 2001.

[14] R. Chen, K. Sivakumar, and H. Kargupta, "Distributed web mining using Bayesian networks from multiple data streams," in *Proceedings of the 2001 IEEE International Conference on Data Mining*, (San Jose, CA), November 2001.

[15] J. Cheng, D. A. Bell, and W. Liu, "Learning belief networks from data: An information theory based approach," in *Proceedings of the Sixth ACM International Conference on Information and Knowledge Management*, 1997.

[16] D. M. Chickering, "Learning equivalence classes of Bayesian network structure," in *Pro-*

ceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence (E. Horvitz and F. Jensen, eds.), Morgan Kaufmann, 1996.

[17] D. M. Chickering and D. Heckerman, "Efficient approximation for the marginal likelihood of incomplete data given a Bayesian network," *Machine Learning*, vol. 29, pp. 181–212, 1997.

[18] G. F. Cooper and E. Herskovits, "A Bayesian method for the induction of probabilistic networks from data," *Machine Learning*, vol. 9, pp. 309–347, 1992.

[19] V. Crestana and N. Soparkar, "Mining decentralized data repositories," Tech. Rep. CSE-TR-385-99, University of Michigan, Ann Arbor, MI, 1999.

[20] S. Dasgupta, "The sample complexity of learning fixed-structure Bayesian networks," *Machine Learning*, vol. 29, pp. 165–180, 1997.

[21] K. J. Ezawa and S. T, "Fraud/uncollectable debt detection using Bayesian network based learning system: A rare binary outcome with mixed data structures," in *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence* (P. Besnard and S. Hanks, eds.), pp. 157–166, Morgan Kaufmann, 1995.

[22] N. Friedman, "Learning Bayesian networks in the presence of missing values and hidden variables," in *Proceedings of the Fourteenth International Conference on Machine Learning* (D. Fisher, ed.), Morgan Kaufmann, 1997.

[23] N. Friedman, "The Bayesian structural EM algorithm," in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence* (G. F. Cooper and S. Moral, eds.), Morgan Kaufmann, 1998.

[24] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine Learning*, vol. 29, pp. 131–163, 1997.

[25] N. Friedman and M. Goldszmidt, "Sequential update of Bayesian network structure," in *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence* (D. Geiger and P. Shanoy, eds.), Morgan Kaufmann, 1997.

[26] N. Friedman and Z. Yakhini, "On the sample complexity of learning Bayesian networks," in *Proceedings of the twelfth conference on uncertainty in artificial intelligence*, 1996.

[27] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov chain Monte Carlo in practice*. Chapman and Hall, 1996.

[28] D. Heckerman, "A tutorial on learning with Bayesian networks," in *Proceedings of the NATO Advanced Study Institute on Learning in Graphical Models* (M. I. Jordan, ed.), Kluwer Academic Publishers, 1998.

[29] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning Bayesian networks: The combination of knowledge and statistical data," *Machine Learning*, vol. 20, pp. 197–243, 1995.

[30] D. Heckerman and D. Gieger, "Learning Bayesian networks: A unification for discrete and Gaussian domains," in *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence* (P. Besnard and S. Hanks, eds.), pp. 274–284, Morgan Kaufmann, 1995.

[31] D. Heckerman, C. Meek, and G. Cooper, "A Bayesian approach to causal discovery," Technical Report MSR-TR-97-05, Microsoft Research, 1997.

[32] D. Hershberger and H. Kargupta, "Distributed multivariate regression using wavelet-based collective data mining," Tech. Rep. EECS-99-02, School of EECS, Washington State University, 1999. To be published in the Special Issue on Parallel and Distributed Data Mining of the Journal of Parallel Distributed Computing, Guest Eds: Vipin Kumar, Sanjay Ranka, and Vineet Singh.

[33] F. Jensen, *An Introduction to Bayesian Networks*. Springer, 1996.

[34] E. Johnson and H. Kargupta, "Collective, hierarchical clustering from distributed, heterogeneous data," in *Lecture Notes in Computer Science*, vol. 1759, pp. 221–244, Springer-Verlag, 1999.

[35] H. Kargupta, W. Huang, S. Krishnamrthy, H. Park, and S. Wang, "Collective principal component analysis from distributed, heterogeneous data," in *Proceedings of the Principles of Data Mining and Knowledge Discovery Conference* (D. Zighed, J. Komorowski, and J. Zytkow, eds.), vol. 1910, (Berlin), pp. 452–457, Springer, September 2000. Lecture Notes in Computer Science.

[36] H. Kargupta, W. Huang, S. Krishnamurthy, and E. Johnson, "Distributed clustering using collective principle component analysis," in *Proceedings of the ACM SIGKDD Workshop on Distributed and Parallel Knowledge Discovery in Databases*, pp. 8–19, August 2000.

[37] H. Kargupta, B. Park, D. Hershberger, and E. Johnson, "Collective data mining: A new perspective toward distributed data mining," in *Advances in Distributed and Parallel Knowledge Discovery* (H. Kargupta and P. Chan, eds.), pp. 133–184, Menlo Park, California, USA: AAAI/ MIT Press, 2000.

[38]H. Kargupta, K. Sivakumar, W. Huang, R. Ayyagari, R. Chen, B.-H. Park, and E. Johnson, "Towards ubiquitous mining of distributed data," in *Data Mining for Scientific and Engineering Applications* (R. Grossman, C. Kamath, P. Kegelmeyer, V. Kumar, and R. Namburu, eds.), Kluwer Academic Publishers, 2001.

[39]Y. Kenji, "Distributed cooperative Bayesian learning strategies," in *Proceedings of the Tenth Annual Conference on Computational Learning Theory*, (Nashville, Tennessee), pp. 250–262, ACM Press, 1997.

[40]R. King and M. Novak, "Supporting information infrastructure for distributed, heterogeneous knowledge discovery," in *Proceedings of SIGMOD 96 Workshop on Research Issues on Data Mining and Knowledge Discovery*, (Montreal, Canada), `http://www.cs.colorado.edu/$\;\tilde{}\;$sanctuary/Papers/datamining.ps`, 1996.

[41]W. Lam and F. Bacchus, "Learning Bayesian belief networks: An approach based on the MDL principle," *Computational Intelligence*, vol. 10, pp. 262–293, 1994.

[42]W. Lam and A. M. Segre, "Distributed data mining of probabilistic knowledge," in *Proceedings of the 17th International Conference on Distributed Computing Systems*, (Washington), pp. 178–185, IEEE Computer Society Press, 1997.

[43]S. L. Lauritzen, "The EM algorithm for graphical association models with missing data," *Computational Statistics and Data Analysis*, vol. 19, pp. 191–201, 1995.

[44]S. L. Lauritzen and D. J. Spiegelhalter, "Local computations with probabilities on graphical structures and their application to expert systems (with discussion)," *Journal of the Royal Statistical Society, series B*, vol. 50, pp. 157–224, 1988.

[45]A. Lazarevic, D. Pokrajac, and Z. Obradovic, "Distributed clustering and local regression for knowledge discovery in multiple spatial databases," in *Proc. 8th European Symposium on Artificial Neural Networks*, (Bruges, Belgium), April 2000.

[46]D. Madigan and A. Raftery, "Model selection and accounting for model uncertainty in graphical models using Occam's window," *Journal of the American Statistical Association*, vol. 89, pp. 1535–1546, 1994.

[47]S. McClean, B. Scotney, and K. Greer, "Clustering heterogeneous distributed databases," in *Workshop on Distributed and Parallel Knowledge Discovery at KDD-2000*, (Boston), pp. 20–29, 2000.

[48]M. Meila and M. I. Jordan, "Estimating dependency structure as a hidden variable," in *NIPS*, 1998.

[49]H. Park, H. Kargupta, E. Johnson, R. E. Sanseverino, L. D. Silvestre, and D. Hershberger, "Distributed, collablorative data analysis from heterogeneous sites using a scalable evolutionary technique," Tech. Rep. EECS98-001, School of Electrical Engineering and Computer Science, Washington State University, 2000.

[50]S. Parthasarathy and M. Ogihara, "Clustering distributed homogeneous datasets," in *Proceedings of the Principles of Data Mining and Knowledge Discovery Conference* (D. Zighed, J. Komorowski, and J. Zytkow, eds.), pp. 566–574, September 2000.

[51]M. Pazzani and D. Billsus, "Learning and revising user profiles: The identification of interesting web sites," *Machine Learning*, vol. 27, pp. 313–331, 1997.

[52]M. Pazzani, J. Muramatsu, and D. Billsus, "Syskill & Webert: Identifying interesting web sites," in *Proceedings of the National Conference on Artificial Intelligence*, 1996.

[53]J. Pearl, *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, 1988.

[54]J. Pearl, "Graphical models. causality and intervention," *Statistical Science*, vol. 8, pp. 266–273, 1993.

[55]A. Prodromidis and S. Stolfo, "Cost complexity-based pruning of ensemble classifiers," in *Workshop on Distributed and Parallel Knowledge Discovery at KDD-2000*, (Boston), pp. 30–40, 2000.

[56]F. J. Provost and B. Buchanan, "Inductive policy: The pragmatics of bias selection," *Machine Learning*, vol. 20, pp. 35–61, 1995.

[57]M. Sayal and P. Scheuermann, "A distributed clustering algorithm for web-based access patterns," in *Workshop on Distributed and Parallel Knowledge Discovery at KDD-2000*, (Boston), pp. 41–48, 2000.

[58]M. Singh, "Learning Bayesian networks from incomplete data," in *Proceedings of the National Conference on Artificial Intelligence*, pp. 27–31, AAAI Press, 1997.

[59]M. Singh and G. M. Provan, "A comparison of induction algorithms for selective and non-selective Bayesian classifiers," in *Proceedings of the Twelfth International Conference on Machine Learning* (A. Prieditis and S. Russel, eds.), pp. 497–505, Morgan Kaufmann, 1995.

[60]D. J. Spiegelhalter and S. L. Lauritzen, "Sequential updating of conditional probabilities on directed graphical structures," *Networks*, vol. 20, pp. 570–605, 1990.

[61]P. Spirtes, C. Glymour, and R. Scheines, *Causation, Prediction and Search*. No. 81 in Lecture Notes in Statistics, Springer-Verlag, 1993.
[62]J. Suzuki, "A construction of Bayesian networks from databases based on an MDL scheme," in *Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence* (D. Heckerman and A. Mamdani, eds.), pp. 266–273, Morgam Kaufmann, 1993.
[63]B. Thiesson, "Accelerated quantification of Bayesian networks with incomplete data," in *Proceedings of the First International Conference on Knowledge Discovery and Data Mining*, pp. 306–311, AAAI Press, 1995.
[64]B. Thiesson, C. Meek, D. M. Chickering, and D. Heckerman, "Learning mixtures of Bayesian networks," in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann, 1998.
[65]A. Thomas, D. Spiegelhalter, and W. Gilks, "Bugs: A program to perform Bayesian inference using Gibbs sampling," in *Bayesian Statistics* (J. Bernardo, J. Berger, A. Dawid, and A. Smith, eds.), pp. 837–842, Oxford University Press, 1992.
[66]K. Tumer and J. Ghosh, "Robust order statistics based ensemble for distributed data mining," in *Advances in Distributed and Parallel Knowledge Discovery*, pp. 185–210, MIT, 2000.
[67]A. Turinsky and R. Grossman, "A framework for finding distributed data mining strategies that are intermediate between centeralized strategies and in-place strategies," in *Workshop on Distributed and Parallel Knowledge Discovery at KDD-2000*, (Boston), pp. 1–7, 2000.
[68]G. Zweig and S. J. Russel, "Speech recognition with dynamic Bayesian networks," in *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, 1998.