Web-based knowledge acquisition to solve inverse problems

Na Tang
Computer Science Dept.
University of California, Davis
Davis, CA 95616

V. Rao Vemuri
Computer Science Dept.
University of California, Davis
Davis, CA 95616

Abstract

Machine learning is the science of building predictors from data randomly sampled from an assumed probability distribution while accounting for the computational complexity of the learning algorithm and the predictor's performance on future data. Much of the work in machine learning is empirical and prone to errors because what the machine learns depends on the adequacy and trustworthiness of the training data. Automated web-based knowledge acquisition can play a useful role in developing systematic methods for solving a class of machine learning problems in which there is insufficient or untrustworthy data. Such problems can be solved by using a regularization procedure - a mathematical strategy that seeks to supply the "missing data." There are several ways of regularizing a problem. Statistical methods, for example, can fill in a few data items. But these methods rely on using the available, and possibly unreliable, data to calculate the missing values. Besides, they perform poorly if the percentage of missing values exceeds a threshold. An alternative is to fill in the missing data by an automated knowledge discovery process via mining the WWW. This novel procedure is applied by first restoring missing information and next learning the structure and parameters of the unknown system from the restored data. Using a Bayesian network as a possible model for the unknown system, the parameters, i.e., the probabilities associated with the causal relationships in the network, are deduced using the knowledge mined from the WWW in conjunction with the data available on hand. The method, when tested against heart disease data sets from the UC Irvine Machine Learning Repository [UCI], gave satisfactory results. Preliminary results of our approach, using the Naive Bayes as the system model, are then compared with the performance of the EM algorithm, a well-understood statistical method. Work is currently in progress to assess the performance of this method in data-poor domains.

1 INTRODUCTION

Inverse problems arise anywhere data is collected which is related to the unknown quantities by a mathematical model. The goal is to estimate the parameters of the model given the input and output. Diagnosis of a disease from symptoms is an inverse problem in medical field. Here, the input and output are given by a training set of symptoms (which can be present, absent or unknown) and diseases, while the parameters of the model to be estimated are the relationships among them. Once the parameters are known, the diseases can be diagnosed given a new patient's symptoms.

One such model is the Bayesian network [Pearl 1988], a probabilistic model successfully applied in many fields. However, training Bayesian networks to make accurate predictions can be difficult or even impossible when there is insufficient information. Today's WWW is a source of huge amount of information, and it can be used to fill the gaps. However, people have to manually get the information (and/or knowledge) and fill in the data because information on the web is understandable only to humans. Existing knowledge discovery techniques can help retrieve the needed information automatically and effectively.

In this paper, a methodology is described to automatically retrieve the relevant knowledge from the web and restore the missing data. Bayesian networks built from the restored data are then used to make predictions. As a result, the web serves as an automated

information resource to build models for systems and applications. The basic assumption here is that the knowledge obtained from the web can correctly reflect the relationship among data items. This assumption is validated in our experiments.

The heart-disease data sets from the UCI Machine Learning Repository [UCI] are used as our initial experimental data to validate our approach. Figure 1 shows a segment from these heart-disease data sets. There are fourteen attributes including the class variable *Outcome* that indicates whether a person has heart disease or not. All the other attributes describe the conditions and symptoms of the patient such as *Age*, *Gender*, *Cholesterol*, etc.

Age	Sex	Chest Pain	Rest BP	Choles-terol	Blood Sugar	ECG	···	Outcome
60	1	3	180	-1	0	1	…	0
60	1	3	120	-1	-1	0	…	1
60	1	2	160	267	1	1	…	1
56	1	2	126	166	0	1	…	0
59	1	4	140	-1	0	1	…	1
62	1	4	110	-1	0	0	…	1
63	1	3	-1	-1	0	2	…	1
63	0	2	-1	-1	0	0	…	0
62	1	4	152	153	0	1	…	1
56	1	2	124	224	1	0		0
…	…	…	…	…	…	…	…	…

Figure 1: "Va" Data Set (one of the four heart disease data sets) from UCI repository. All "-1"s stand for missing values.

A good Bayesian model can be built based on complete training data sets and can then be used to make a diagnosis, given one's conditions. Methods of building Bayesian networks from complete data, such as Naive Bayes (NB) and Tree Augmented Naive Bayes (TAN), are discussed in [Friedman&Goldzmidt 1997, Heckerman 1995]. A NB model assumes that attributes are independent from each other. Figure 2 shows a NB model of the interdependence between heart disease and a number of factors. A TAN model has more complex topology, which is obtained by seeking through some possible network structures. Both NB and TAN can retrieve satisfactory prediction results in the presence of complete data. However, in many practical circumstances, data entries of some attribute may be partially missing or completely missing. An attribute whose values are completely missing is called an *incomplete attribute*. *Incomplete attributes* can arise, (in the heart disease problem), for example, when some recently acquired new knowledge points to the possibility of a newly discovered cause to a disease condition.

The rest of paper is organized as follows. We review some related work in Section 2. In Section 3, we describe our approach to fill in missing data for constructing Bayesian networks. Experimental results are reported in Section 4. We conclude and discuss our future work in Section 5.

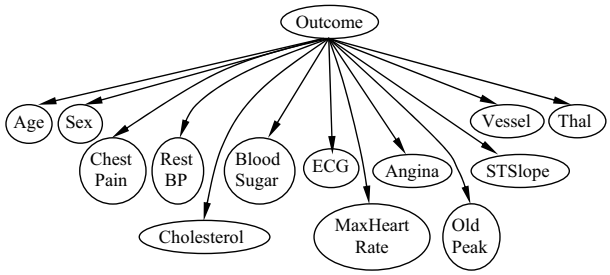


Figure 2: Naive Bayes Model for heart-disease data

2 RELATED WORK

Several statistical methods are available to construct Bayesian networks from incomplete data. Some of them are described below:

(1) A simple method is to fill in missing data values using available data as a guide [Spiegelhalter 1990]. A missing entry can be filled in with the most probable value of the *incomplete attribute* or with the most probable value in the corresponding predicting class. An alternative way is to find an attribute which the *incomplete attribute* is highly dependent on and then estimate the missing value according to the observed value in the obtained attribute. This method is not feasible when the values of an attribute are completely missing (i.e. an entire column).

(2) Expectation-maximization (EM) algorithm [Dempster 1977] estimates the parameters by iteratively finding the expectation of a parameter and then finding the maximum likelihood estimate (MLE) using the parameter from the expectation step. It provides more reliable estimates of the parameters than (1) in the presence of missing data. However, it relies on the assumption that data items are missing at random (MAR): That is, with each configuration, the available data is a representative sample of the complete data, which is not true in our case because sometimes the available data can not reflect the relationship between the *incomplete attribute* (whose values are totally missing) and the values of the other known attributes. Gibbs Sampling (GS) [Geman 1984], another sophisticated statistical method, makes the same MAR assumption. Therefore, the details are omitted here.

(3) Other statistical methods seek to simultaneously fill in the missing data while searching for an optimal structure of the Bayesian network model. Two of these methods are briefly outlined here. Structural

EM [Friedman 1999, Friedman 1997] stars an initial structure and passes the structure to the EM algorithm. The MLE returned by EM is considered as the score for the structure. A new structure is generated by adding, deleting or reversing an edge in the previous structure and the new structure is passed to the EM algorithm again and the score for the new structure is returned and compared to the previous score. The process is repeated until there is no improvement for the score. The Evolutionary Algorithm (EA) [Myers 1999], uses genetic algorithm to evolve both network structures and missing values to find an optimal Bayesian network.

One limitation of these methods is that they lead to not-so-accurate predictions when a large percentage of data is missing or when data is non-randomly missing. For example, EA finds good predictive network at 5%, 10% and 15% missing data while the predictive accuracy degrades sharply at 30% missing data. To address the limitations, a novel approach is proposed in this paper to deduce the missing information via a knowledge discovery process in conjunction with mining the WWW.

Some of the other papers in web mining research are closer in concept to the research presented in this paper. Reference [Craven 2000] shows a methodology for extracting useful information to fill a knowledge base. This method, WWW→KB, retrieves relevant documents from certain web sites via Naive Bayes text classifier, then extracts useful symbolic information from the retrieved documents via an information extraction technique called SRV (Sequential Rules with Validation), and finally builds the knowledge base from the extracted data. In this paper this idea is extended to WWW→KB→BN in the sense that the knowledge extracted from the web is used to help construct Bayesian networks (BN) from incomplete data. Also, the method differs from the WWW→KB method in three important respects. First, the method described here uses search engines to search a large portion of the WWW whereas the WWW→KB method performs content mining within a specified context. Two, the proposed method extracts probabilistic information to fill in missing values rather than symbolic information. Three, the proposed method performs further processing to handle the uncertainty associated with the extracted information.

Web query systems [Kwok 2001, Lam 2002] are also closely related to our mining approach. They analyze the queries, use search engines to search the web, and employ information retrieval and extraction techniques to get the answers to user queries. However, their major goal is to answer queries. Therefore, they focus on the query categorization. Furthermore, most of the sought-after information is very simple and directly resides in the text, while the knowledge needed in this paper is implicitly indicated by the text and requires further processing to be converted to probabilities.

3 RESTORING MISSING DATA VIA KNOWLEDGE DISCOVERY FROM WEB

The crux of the idea here is to look for patterns of relationship between the *incomplete attribute* and other available attributes of the problem. It is assumed that the sought-after relationship information appears in some web document either as a natural language sentence or as an item in a table. In either case the relationship indicates how the available attributes influence the *incomplete attribute* or how the *incomplete attribute* influences the other attributes. The implementation architecture is shown in Figure 3. For simplicity, three-way relationships and cascade relationships are not examined in this paper; only binary relationships (i.e., relationships between two attributes at a time) are considered to make the classification step (Step C1) and extraction process (Step C2) simple to handle.

The three steps C1, C2 and C3 are explained in the following subsections. It is assumed that *Cholesterol* and *RestBP* are two *incomplete attributes* in the heart data table. *Cholesterol* is the attribute indicating people's cholesterol values and *RestBP* is the attribute indicating people's resting blood pressure values.

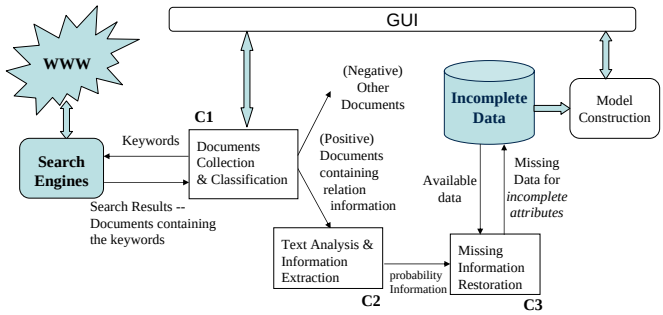


Figure 3: Implementation architecture for generating missing information

3.1 STEP C1: DOCUMENTS COLLECTION AND CLASSIFICATION

Document Collection Phase: N (= 300 or more, typically) documents are collected from Google using a compound search phrase comprised of one *incomplete attribute* and one completely specified attribute. For

example, if *Cholesterol* is an *incomplete attribute*, the probability information between *Cholesterol* and other attributes would be of interest. To extract documents about *Cholesterol* and *Outcome*, we use the keyword set: "cholesterol heart disease" (Similarly, the keyword set "cholesterol age" is used to get documents containing relationship information between *Cholesterol* and *Age* and so on.) Google is chosen because it is a widely available search engine; it can obtain all sorts of topics making it a vehicle suitable for many fields. A specialized search engine and a specialized database, such as Medline, may work better for a specialized domain, say, the health-care field.

Document Classification Phase: A trained Naive Bayes Classifier (Rainbow toolkit [McCallum 2002]) is used to divide the collected documents into two classes: the positive class containing information on the causes of heart-disease and associated probability data and the negative class comprising of all other documents. As the documents in the positive class (positive documents) contain the needed probability information, they are retained for further processing; the negative documents are discarded. Naive Bayes text classifier [Mitchell 1997] is a probabilistic classifier that calculates the probability of a document belonging to each possible class and then assigns the document the class with the highest probability.

The Naive Bayes text classifier needs to be trained before the preceding step can be implemented. This training is performed manually as follows. Using a number of professional web sites that specialize in heart disease [Heart websites], 200 documents are collected, manually inspected and hand-labeled as positive and negative (50 for the positive class and 150 for the negative class). Naive Bayes text classifier is used here because of its high accuracy and low complexity. It outperforms k-Nearest Neighbors (k-NN) in our categorization task.

3.2 STEP C2: TEXT ANALYSIS AND INFORMATION EXTRACTION

Only HTML (Hyper-Text Markup Language) or plain text files on the web are examined in this paper because they are the most popular format on the WWW. The text classifier and extraction rules described in this paper can only deal with such format. The embedded knowledge may reside in free text as well as structured text (e.g. tables). For the latter, some HTML tags become very useful. Rules are devised to process and extract the probability information. Each sentence and each table from the retrieved documents (i.e., the positive documents) is analyzed and examined to see if they matched any of the rules. The outputs of the probability information are collected for processing Step C3.

The probability information targeted for extraction, i.e. the relations between the *incomplete attribute* and other attributes, is divided into two categories: *point probabilities* and *qualitative influences*. Other forms of probability information such as *comparison* and *qualitative synergies* would also be useful but are not considered in this paper. Formal definitions of all these items are given in [Druzdzel 1995]. Only the above two categories are discussed because of their simplicity. The extraction rules are designed to extract these two types of probability information, which are based on a group of attributes including sentence length (or table length), vocabulary for heart disease, distance between words, and so on.

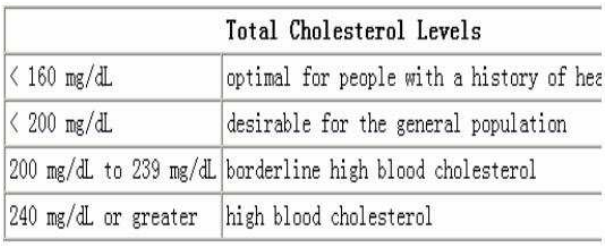
3.2.1 Point probabilities

Point probabilities can be expressed in the mathematical form $P(a_i|a_j1, ...a_jk) = c$, where c is a constant. As mentioned before, to simplify the extraction task, we only examine $P(a_i|a_j)$, i.e. the probabilistic relationship between two attributes instead of many attributes. Take the relation between *Cholesterol* and *Outcome* as an example. The degree of risk for heart disease for different levels of cholesterol is usually explicitly described in the relevant documents. For example, the probability information indicated by the text in Figure 4 (a) is that $P(Outcome = 1|Cholesterol < 160mg/dl)$ is very low, $P(Outcome = 1|Cholesterol < 200mg/dl)$ is low, $P(Outcome = 1|200mg/dl < Cholesterol < 239mg/dl)$ is borderline high and $P(Outcome = 1|Cholesterol > 240mg/dl)$ is very high.

In the documents examined, we found that the *point probabilities* are typically expressed in two ways: (a) by the use of tables (structured text, see **Example 1**) and (b) by the use of regular sentences (free text, see **Example 2**).

Example 1: Figure 4 illustrates an example of how the relation between *Cholesterol* and *Outcome* appears in a table and the rule to extract useful information from tables. If the table matches the regular expression defined in the rule, we extract the cholesterol levels and the degree of heart-disease risk.

Vocabulary about heart disease in the extraction rules needs to be defined. For example, *LevelOfHeartDisease* can be "optimal", "ok", "borderline", "high" etc., which indicates the degree of heart-disease risk, i.e., the probability of having heart disease. *NameOfIncompleteAttribute* and *UnitOfIncompleteAttribute* are "blood pressure" and "mm Hg" respectively if the relation between *RestBP* and other attributes is to be examined.



Total Cholesterol Levels	
< 160 mg/dL	optimal for people with a history of hea
< 200 mg/dL	desirable for the general population
200 mg/dL to 239 mg/dL	borderline high blood cholesterol
240 mg/dL or greater	high blood cholesterol

(a)

```
Table =  TableStartTag * TableEntryTag * level * degree * TableEndTag
         | TableStartTag * TableEntryTag * degree * level * TableEndTag,
Length(table) < 1500 ,
Contains(NameOfIncompleteAttribute, Table) = True,

   TableStartTag = "<table*>", TableEndTag = "</table>",
   TableEntryTag = "<tr*>",
   level = LeftOp Number (UnitOfIncompleteAttribute | "")
         | Number (UnitOfIncompleteAttribute | "") MidOp Number
         (UnitOfIncompleteAttribute | "")
         | Number (UnitOfIncompleteAttribute | "")  RightOp,
   degree    = LevelsOfHeartDisease
   LeftOp    = "less than" | "lower than" | "below" | "under" | "<"
             | "greater than" | "over" | ">"
   MidOp     = "to" | "-"
   RightOp   = ("and" | "or") ("lower" | "less" | "below" | "more" | "greater"
             | "higher" | "above")

Output: (level, degree)
```

(b)

Figure 4: *Point probabilities* expressed in tables and extraction rule: (a) Text that appears in the web browser; (b) Extraction Rule. Here "*" stands for any character and "|" for "or".

The extracted output from the table is ("< 160 mg/dl", "optimal"), ("< 200 mg/dl", "desirable"), ("200 mg/dL to 239 mg/dl", "borderline high") and ("240 mg/dl or greater", "high"). The output (level, degree) can be interpreted in probability format as $P(Outcome = 1|Cholesterol \in level) = degree$.

Example 2: An example of probabilities expressed by regular sentences (free text) is given below.

"In general, total cholesterol is considered high when 240 or more, borderline when 200-239, and desirable when 200 or less."

We extract cholesterol levels and the degree of heart-disease risk by using a procedure similar to the one used in **Example 1**. The output for the sentence above is ("200 or less", "desirable"), ("200-239", "borderline") and ("240 or more", "high"). The output (level, degree) can be converted to probability information in the same way with the table extraction. The words ("desirable", "borderline" ...) that describe the probability are finally converted into numerical values.

3.2.2 Qualitative Influences

Qualitative influences describe how one attribute influences another in a qualitative way. It consists of positive and negative influence. A positive influence from attribute A_i to A_j means that choosing a higher value for A_i makes the higher value for A_j more likely. A negative influence is defined in a similar way.

Example 3: An example of extracting information about *qualitative influences* is given below.

> "As people get older, their cholesterol levels rise."
> "Cholesterol levels naturally rise as men and women age."
> "Women have lower total cholesterol levels than men of the same age."

The extraction rule for the *qualitative influences* is defined in a similar way as the *point probabilities*. The output of the first two confirms a positive influence from *Age* to *Cholesterol* and the output of the fourth confirms a positive influence from *Gender* to *Cholesterol* (here, we assume "women" < "men" as the order of the attribute *Gender*).

It is assumed that the probability information published in most web sites is the most reliable information. For example, if most web sites show that it is optimal to have cholesterol less than 200mg/dl while a few web sites regard 160mg/dl as the separation line, then the former is chosen. In addition, the obtained information come from the top of the list returned by the search engine. It is implicitly assumed that the higher-ranked search results are more reliable.

3.3 STEP C3: MISSING INFORMATION RESTORATION

There are two types of output from Step C2: *point probabilities* and *qualitative influences*. For example, the *point probabilities* for the relation between *Outcome* and *Cholesterol* can be represented by:

$$P(\text{Outcome} = 1|\text{Cholesterol} = v_1) = v_2, \qquad (1)$$

where v_1 stands for a range of cholesterol levels and v_2 is a constant. The second type of output from Step C2 includes a positive influence from *Age* to *Cholesterol* and a positive influence from *Gender* to *Cholesterol*. A positive influence from *Age* to *Cholesterol* means that with lager value of *Age* the risk of getting higher values of *Cholesterol* is greater. This fact can be represented by:

$$P(\text{Cholesterol} > v|\text{Age} = a_1) > P(\text{Cholesterol} > v|\text{Age} = a_2) \quad (2)$$

given $a_1 > a_2$.

A positive influence from *Gender* to *Cholesterol* can be interpreted in similarly. Given all these probability outputs from Step C2, and given the probability constrains such as P(*Age*, *Gender*, *Outcome*) = v from the available data, we can elicit the probabilities P(*Cholesterol* | *Age*, *Gender*, *Outcome*) based on the approach described in [Druzdzel 1995]. This method allows us to convert all the probability information into a linear system of equalities and inequalities, from which bounds on the probabilities of interest are calculated. From these bounds, it is possible to elicit the required probabilities, namely, $P(Cholesterol|Age, Gender, Outcome)$. Now the missing values in the data set can be filled in based on these probabilities. For example,

$$P(Cholesterol < 200|pred) = v_1,$$
$$P(200 < Cholesterol < 240|pred) = v_2,$$
$$P(Cholesterol > 240|pred) = v_3,$$
$$pred = \text{“}Age < 50, Gender = female, Outcome = 1\text{”}.$$

If a female patient is younger than 50 and has heart disease, then we set her missing cholesterol number to one of the values from the set < 200, $200 - 240$, > 240 with probabilities $\frac{v_1}{v}$, $\frac{v_2}{v}$, and $\frac{v_3}{v}$, respectively, where $v = v_1 + v_2 + v_3$.

4 EXPERIMENTAL RESULTS

We conducted our experiments on the Cleveland data set, which is a complete data set, i.e. a table with no missing values. We randomly chose a portion of the data for training and the remaining for testing. First, we trained our Naive Bayes model and TAN model using the complete training data and obtained the heart-disease prediction accuracies with the testing data. Then we assumed that all the *Cholesterol* values or *RestBP* values or both in the training data were missing and applied our method to fill in these values. Figure 5 shows the fill-in accuracies with different training data size for *Cholesterol* and *RestBP*. The fill-in accuracy of *Cholesterol* is defined as

$$\frac{\sharp \text{ of correct filled-in Cholesterol values}}{\text{total } \sharp \text{ of filled-in Cholesterol values}}$$

and it is similar to *RestBP*. It shows that the knowledge extracted from the web is capable of restoring incomplete data. The fill-in accuracy of *Cholesterol* is better than that of *RestBP* because less knowledge of *RestBP* is extracted from the web. For example, the relationship of gender and blood pressure is not found. Also, the fill-in accuracy increases as the training data size gets bigger.

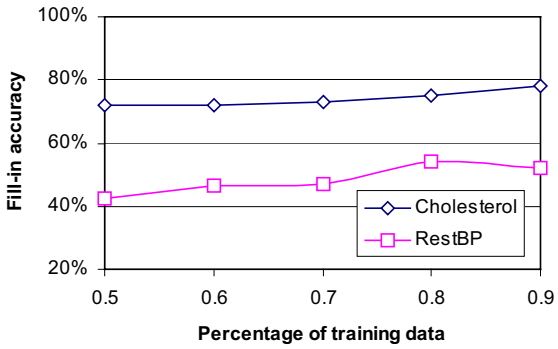


Figure 5: Fill-in accuracies with different training data size.

Based on the training data with the filled-in values, we built our NB and TAN models and evaluate the trained networks with the testing data. We also tested the prediction accuracies without considering the *incomplete attributes*. To compare our approach with the statistical methods, we applied EM algorithm to estimate the NB parameters by considering the missing attributes as hidden variables to test the prediction accuracies. All the heart-disease prediction accuracies are shown in Figure 6 with missing *Cholesterol*, missing *RestBP*, or both, respectively.

The results show that the incomplete data (the data without *incomplete attributes*) can deteriorate the NB and TAN models, i.e. it leads NB and TAN to lower prediction accuracies than the original complete data. Generally the incomplete data has greater influence on the TAN model than on the NB model. An increment of missing data results in a decrease of the prediction accuracy. The EM algorithm can only improve the prediction accuracy in a very small degree. The filled-in data set via our approach outperforms the EM algorithm and performs almost as good as or sometimes even better than the original data. It works better with the TAN model than with the NB model. Also, the percentage of missing data does not have an influence on the performance of the filled-in data with either model. This is because the missing data is filled in based on the extracted knowledge but not on the available data only. Therefore, it can be a promising method when a large percentage of data is missing.

5 SUMMARY AND DISCUSSION

In our study, the web was used as the source for gathering useful information to restore incomplete relational

tables. A number of different techniques in different areas were involved. We mined the web using search engines, selected the relevant documents via categorization techniques, extracted the useful knowledge via information extraction techniques, and finally filled in the missing information via uncertainty processing. The filled-in tables were, in turn, used to simulate two Bayesian network models of the unknown system under study.

Several interesting questions need to be addressed before this method can be effectively used to solve realistic problems. The most obvious questions pertain to the performance (including computational complexity), reliability and scalability of the method. The Naive Bayes text classifier used in Step C1 is a supervised learning method and thus takes some manual effort of collecting and labeling training documents. An alternative would be unsupervised document categorization (i.e., document clustering). Currently work is in progress to investigate the potential of existing clustering techniques. More sophisticated information extraction techniques are also being examined to be used in Step C2 to reduce the complexity. Reliability depends on the confidence one can place on the filled-in numbers generated by this method. That is, any possible confliction between the extracted information and the available data need to be further examined. Scalability refers to the range of missing data. We are particularly interested in the case when there is a large percentage of missing data, i.e., when the problem domain is data poor.

References

M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery (2000). Learning to Construct Knowledge Bases from the World Wide Web. *Artificial Intelligence*, 118(1/2):69–113.

A. Dempster, N. Laird, and D. Rubin, (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society*, B39:1-38

M. J. Druzdzel, L. C. van der Gaag (1995). Elicitation of Probabilities for Belief Networks: Combining Qualitative and Quantitative Information. *Eleventh Annual Conference on Uncertainty in Artificial Intelligence* (UAI-95), 141-148, Morgan Kaufmann Publishers, Inc., San Francisco, CA.

N. Friedman (1999). The bayesian structural em algorithm. *Fourteenth Conference on Uncertainty in Artificial Intelligence*, 647-654.

N. Friedman (1997). Learning Belief Networks in the Presence of Missing Values and Hidden Variables. *Fourteenth International Conference on Machine Learning*, 125-133.

N. Friedman and M. Goldzmidt (1997). Bayesian Network Classifiers. *Machine Learning*, 29:131-163.

S. Geman and D. Geman (1984). Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721-741.

Heart websites:
http://www.heartinfo.org,
http://www.americanheart.org,
http://www.heartcenteronline.com,
http://www.heartsavers.org,
http://www.healthandage.com,
http://www.nhlbi.nih.gov,
http://heartdisease.about.com,
http://www.medem.com/medlb/medlib_entry.cfm.

D. Heckerman, D. Geiger and D. M. Chickering (1995). Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. *Machine Learning*, 20(3):197-243.

C. C. T. Kwok, O. Etzioni, and D. S. Weld. (2001). Scaling question answering to the web. *World Wide Web*, 150-161.

S. K. S. Lam and M. Tamer Özsu. (2002). Querying web data - the webqa approach. *Third International Conference on Web Information Systems Engineering*(WISE 2002), 139-148.

T. M. Mitchell. (1997). *Machine Learning.* McGraw-Hill, New York.

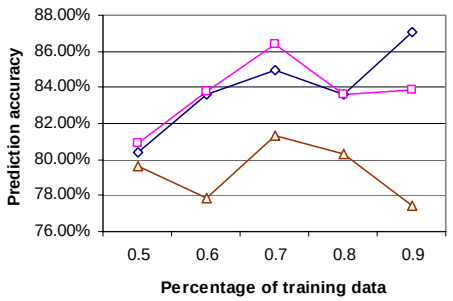
J. W. Myers, K. B. Laskey and K. A. DeJong (1999). Learning Bayesian Networks from Incomplete Data Using Evolutionary Algorithms. In Wolfgang Banzhaf, Jason Daida, Agoston E. Eiben, Max H. Garzon, Vasant Honavar, Mark Jakiela, and Robert E. Smith, editors, *Genetic and Evolutionary Computation Conference*, 1:458-465, Or-lando, Florida, USA, Morgan Kaufmann.

J. Pearl (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers, Inc., San Mateo, CA.

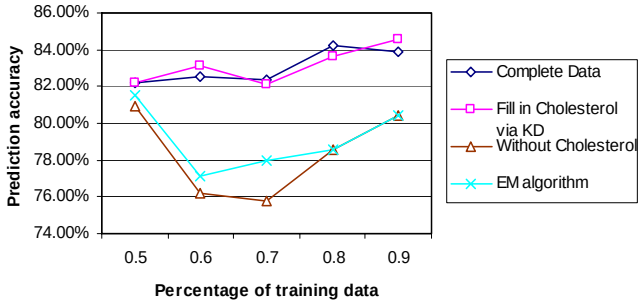
A. McCallum (2002). Rainbow Library (in C language). http://www-2.cs.cmu.edu/∼mccallum/bow/rainbow/

D. Spiegelhalter and S. Lauritzen (1990). Sequential Updating of Conditional Probabilities on Directed Graphical Structures. *Networks*, 20:579-605.

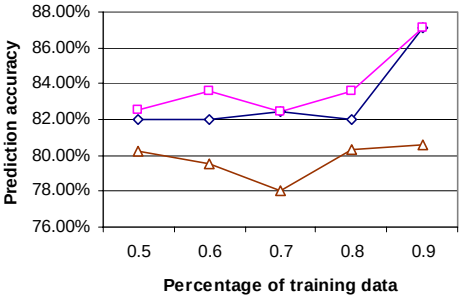
UCI Machine Learning Repository.
http://www.ics.uci.edu/∼mlearn/MLRepository.html



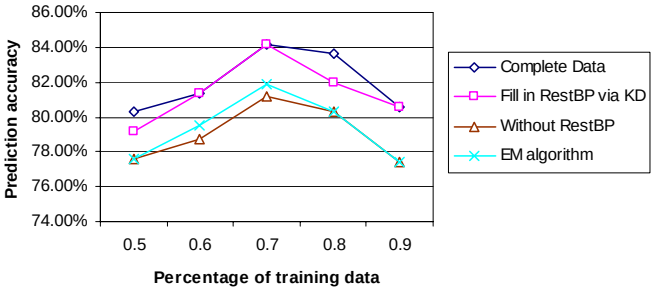
(a) Heart-diseases predictions with missing *Cholesterol* via TAN



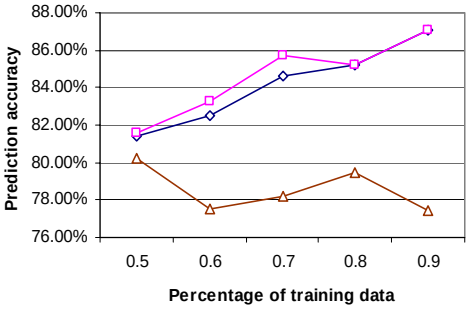
(b) Heart-diseases predictions with missing *Cholesterol* via NB



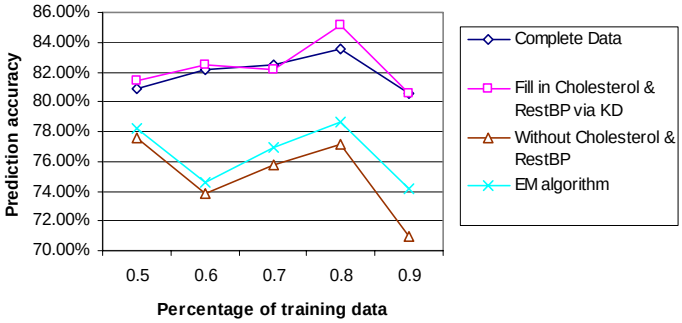
(c) Heart-diseases predictions with missing *RestBP* via TAN



(d) Heart-diseases predictions with missing *RestBP* via NB



(e) Heart-diseases predictions with missing *Cholesterol* & *RestBP* via TAN



(f) Heart-diseases predictions with missing *Cholesterol* & *RestBP* via NB

Figure 6: Heart-disease prediction accuracies with missing *Cholesterol* (a, b), missing *RestBP* (c, d) or missing *Cholesterol* & *RestBP* (e, f)