Does Query-Based Diagnostics Work?

Parot Ratnapinda[1] and **Marek J. Druzdzel**[1,2]
par34@pitt.edu, marek@sis.pitt.edu
[1] Decision Systems Laboratory, School of Information Sciences and Intelligent Systems Program,
University of Pittsburgh, Pittsburgh, PA 15260, USA
[2] Faculty of Computer Science, Białystok University of Technology, Wiejska 45A, 15-351 Białystok, Poland

Abstract

Query-based diagnostics (Agosta, Gardos, & Druzdzel, 2008) offers passive, incremental construction of diagnostic models that rest on the interaction between a diagnostician and a computer-based diagnostic system. Effectively, this approach minimizes knowledge engineering, the main bottleneck in practical application of Bayesian networks. While this idea is appealing, it has undergone only limited testing in practice. We describe a series of experiments that subject a prototype implementing passive, incremental model construction to a rigorous practical test. We show that the prototype's diagnostic accuracy reaches reasonable levels after merely tens of cases and continues to increase with the number of cases, comparing favorably to state of the art approaches based on learning.

1 Introduction

Even though Bayesian network (BN) models (Pearl, 1988) have proven useful in diagnostic domains, they are quite hard to field in practice. Interestingly, it is not computational complexity that is critical here. The main hurdle in applying Bayesian networks to complex diagnostic problems seems to be model building.

One way of addressing this problem is learning models from data accrued over past cases. Given a sufficiently large set of past cases, we can learn both the structure and the parameters of Bayesian networks (Cooper & Herskovits, 1992; Pearl & Verma, 1991; Spirtes, Glymour, & Scheines, 1993). Although model construction from data can significantly reduce knowledge engineering effort, learning faces other problems, such as small data sets, unmeasured variables, missing data, and selection bias. Collections of past cases that are large and complete enough are often hard to find. There are many complex devices that do not break too often or, at least, are not supposed to break often. When dealing with some devices, it is not uncommon to spend months on constructing models that become outdated soon after deployment. Building Bayesian networks requires such a considerable effort on the part of knowledge engineers and domain experts that it is considered the main bottleneck in this area.

There have been several lines of research outside of learning from data that focus on model building. The first approach focuses on providing more expressive building tools. The Noisy-OR model (Pearl, 1988; Henrion, 1989) and its generalizations (Díez, 1993; Srinivas, 1993) simplify the representation and elicitation of independence interactions among multiple causes. Heckerman (1990) developed the concept of *similarity networks* in order to facilitate structure building and probability elicitation. The second approach, usually referred to knowledge-based model construction (KBMC), emphasizes aiding model building by automated generation of decision models from a domain knowledge-base guided by the problem description and observed information (see a special issue at the journal IEEE Transactions on Systems, Man and Cybernetics on the topic of KBMC (Breese, Goldman, & Wellman, 1994)). The third approach is to apply system engineering and knowledge engineering techniques for aiding the process of building Bayesian networks. Laskey and Mahoney (1996; 1997) address the issues of modularization, object-orientation, knowledge-base, and evaluation in a spiral model of development cycle. Koller and Pfeffer (1997; 1999) developed Object-Oriented Bayesian Networks (OOBN) that use objects as organizational units to reduce the complexity of modeling and increase the speed of inference. Lu *et al.* (2000) propose mechanism-based model construction, in which models are constructed from a collection of mechanisms based on scientific laws or pieces of existing models. (Ibargengoytia, Vadera, & Sucar, 2006) propose to learn a Bayesian network model for a normal mode of operation, for which data are typically available, and then detect anomalies as deviations from this model.

Agosta *et al.* (2008) went further and proposed an approach that eliminates knowledge engineering altogether. In what they call query-based diagnostics, they propose embedding a diagnostic aid in existing systems for diagnostic record keeping. A diagnostician working on a case, recording symptoms and other findings along with the final diagnosis, without being aware of it, participates in constructing a simplified Bayesian network model that supports future cases. From the theoretical perspective, the idea is a combination of structure elicitation and incremental learning. The diagnostician provides the system with a basic distinction between symptoms, background information, and the final diagnosis. Past cases solved by diagnosticians can provide considerable information about the domain. Every new case acquired by the system adds useful information and, in the long run, leads to building a usable model. As cases accrue, the system refines the structure and the parameters of such model and improves its accuracy.

While this idea is appealing, it has undergone only limited testing in practice. To the best of our knowledge, there are two existing prototypes implementing this approach. An industrial prototype of the system has been implemented and fielded at Intel and tested in the domain of diagnostics and corrective maintenance of factory equipment (Agosta, Khan, & Poupart, 2010). A widely accessible prototype, called Marilyn (Pols, 2007), was tested in a limited setting of a help desk at a university computing laboratory (Ratnapinda & Druzdzel, 2009). Neither of the two prototypes and the very idea of a system that eliminated completely the knowledge engineering phase and learns successively from diagnostic cases have undergone a formal evaluation. In this paper, we attempt to evaluate one of these two prototypes (Marilyn) systematically, based on several real data sets, obtained from the Irvine Machine Learning Repository.[1] We show that the prototype's diagnostic accuracy reaches reasonable levels after merely tens of cases and continues to increase with the number of cases, comparing favorably with state of the art approaches based on learning.

2 Background

We will start with a brief review of the technology involved in a query-based diagnostic prototype like Marilyn, notably Bayesian networks, noisy-OR gates, and the EM algorithm.

2.1 Bayesian Networks

Bayesian networks (Pearl, 1988) are acyclic directed graphs representing joint probability distributions over sets of variables. Every node is the graph represents a random variable. Lack of an arc between two nodes represents conditional independence between the variables that these nodes represent. Nodes are quantified by means of conditional probability tables (CPTs), representing the probability distribution of the variables that they represent conditional on their parent variables in the graph. Nodes without parents are specified by prior probability distributions. The joint probability distribution over a set of variables $\mathbf{X} = \{X_1, \ldots, X_n\}$ can be obtained by taking the product of all prior and conditional probability distributions:

$$\Pr(\mathbf{X}) = \Pr(X_1, \ldots, X_n) = \prod_{i=1}^{n} \Pr(X_i | Pa(X_i)) \ . \quad (1)$$

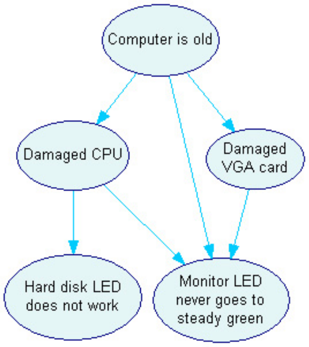

Computer is old
Damaged CPU
Damaged VGA card
Hard disk LED does not work
Monitor LED never goes to steady green

Figure 1: An example Bayesian network modeling computer hardware problems

Figure 1 shows a simple Bayesian network modeling two computer hardware problems. The variables in this model are: *Computer is old*, *Damaged CPU*, *Damaged VGA card*, *Hard disk LED does not work* and *Monitor LED never goes to steady green*. Each of the variables in the model is binary, i.e., has two outcomes: *True* and *False*.

A directed arc between *Damaged CPU* and *Hard disk LED does not work* indicates that *Damaged CPU* will affect the probability that *Hard disk LED does not work*. Similarly, an arc from *Computer is old* to *Damaged VGA card* indicates that computer age influences the likelihood of a damaged VGA card.

The most important type of reasoning in Bayesian networks is known as belief updating and amounts to computing the probability distribution over variables of interest given the evidence. For example, in the model of Figure 1, the variable of interest could be *Damaged*

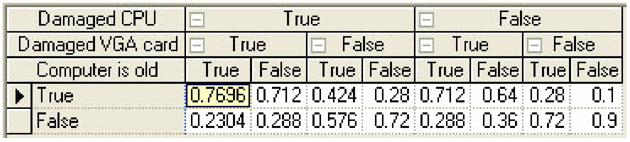
[1] http://archive.ics.uci.edu/ml/

CPU and the BN could compute the posterior probability distribution over this node given the observed values of *Computer is old*, *Hard disk LED does not work*, and *Monitor LED never goes to steady green.* Once the network has updated the probability values, these can be used to make a diagnostic decision.

2.2 The Leaky Noisy-OR Gate

Bayesian networks suffer from a practical problem: Because CPTs represent the probability distribution of a node conditional on all combinations of parent variables, their size grows exponentially with the number of parents. Table 1 shows the CPT for the node *Monitor LED never goes to steady green.* The node has three parents and the size of its CPT is $2^3 = 8$.

Table 1: Conditional probability table of the node *Monitor LED never goes to steady green*



Damaged CPU	True				False			
Damaged VGA card	True		False		True		False	
Computer is old	True	False	True	False	True	False	True	False
True	0.7696	0.712	0.424	0.28	0.712	0.64	0.28	0.1
False	0.2304	0.288	0.576	0.72	0.288	0.36	0.72	0.9

One solution to the exponential growth of CPTs is application of Independence of Causal Influences (ICI) models (Díez & Druzdzel, 2006). The ICI models assume that parent variables can cause the effect independently of each other. This assumption allows to reduce the number of parameters needed to specify an interaction from exponential to linear in the number of parents.

Marilyn is based on the ICI model called the noisy-OR gate (Pearl, 1988; Henrion, 1989). The noisy-OR gate is a probabilistic extension of the deterministic OR gate. Each variable in a noisy-OR gate is binary and has two states: *present* and *absent.* Presence of the parent variables X_i effects the presence of the child variable Y. If all the parent variables are *absent*, then the child variable is also *absent.*

In general, it is infeasible to explicitly include all possible causes of an effect. Marilyn uses an extension of the noisy-OR gate called *leaky noisy-OR* gate (Henrion, 1989; Díez, 1993). The parameter p_i of a leaky noisy-OR gate is defined as the probability that Y will be true if X_i is present and every other parent of Y, including unmodeled causes of Y (the leak), are absent.

2.3 The EM Algorithm

The Expectation-Maximization (EM) algorithm (Dempster, Laird, & Rubin, 1977) is widely used to compute maximum likelihood estimates given incomplete data. Implementations of the EM algorithm have been successfully applied to compute parameters of ICI models, including the noisy-OR model (Natarajan et al., 2005; Vomlel, 2006; Abbasi, Dailey, Afzulpurkar, & Uno, 2010). EM consists of two steps: (1) the expectation step (E-step) uses current parameters to compute the expected values of the missing data, and (2) the maximization step (M-step), in which the maximum likelihood of the parameters are estimates based on the current expected values of the data. Then, the EM process repeats until it converges to the maximum likelihood.

3 Marilyn

Marilyn is a web-based application that implements the idea of query-based diagnostics, i.e., passive construction of diagnostic decision models. It is written in C# and ASP.NET, using a Microsoft SQL database to store data. It utilizes the Bayesian reasoning engine SMILE[2] running under the Microsoft Windows Vista Server. Figure 2 shows Marilyn's architecture. Marilyn appears to the user diagnostician as a computer program for logging case data. The user interacts with it though a web browser, entering elements of the case at hand. The case data are entered in free text format, although the system performs simple text matching to suggest values entered in prior cases. Marilyn presents the user unobtrusively with a list of most likely diagnoses implied by the observations entered so far, suggests additional observations to make and tests to perform. Behind the screen, Marilyn constructs a Bayesian network from the prior cases stored in the database and, ultimately, adds the current case to the database.

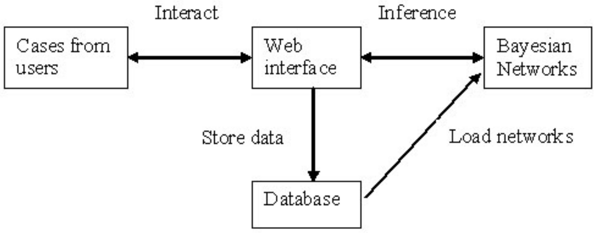


Figure 2: Marilyn's architecture

3.1 Model Structure

The Bayesian networks constructed by Marilyn use a simplified structure called the BN3M model (Kraaijeveld & Druzdzel, 2005), which distinguishes three fundamental types of variables:

[2] http://genie.sis.pitt.edu/

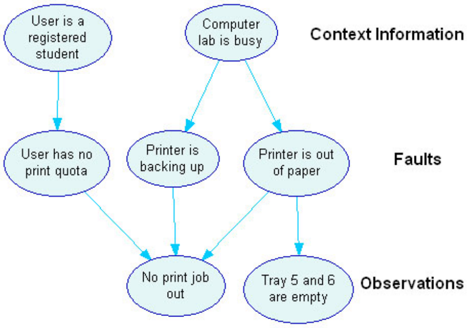


Figure 3: An example of a BN3M model

- *Fault* variables, which represent the problems that the diagnostician wants to identify (e.g., a disease or a device malfunction).
- *Observation* variables, which include observed symptoms and results of diagnostic tests.
- *Context* variables, which are the background, history, or other information known by the technician performing the diagnosis that may influence the probability of a fault and, therefore, are relevant to the diagnosis.

The structure of BN3M networks consists of three levels, with the *context information* variables on the top, the *fault* variables in the middle, and the *observation* variables at the bottom. Influences are possible only between neighboring layers. Figure 3 shows an example of this structure. The first context variable, *User is a registered student*, influences the variable *User has no print quota*. The second context variable *Computer lab is busy* influences the faults *Printer is backing up* and *Printer is out of paper*. *No print job out* is influenced by any three of the fault variables. *Trays 5 and 6 are empty* is influenced only by the fault *Printer is out of paper*.

3.2 Model Construction in Marilyn

When Marilyn starts, it constructs a Bayesian network from the existing database (in the very beginning, this database is empty). The database consists of six tables: *arcs*, *diagnosis*, *domains*, *nodes*, *lablog*, and *emlog*. The first four tables store the information about causal interactions among variables, the number of diagnostic sessions that have been stored by the system, the diagnostic domains, and variables, respectively. The last two tables store data for each session and store the diagnostic logs used in refining the model parameters.

Marilyn constructs the BN3M structure by going through all diagnostic cases entered so far and connecting all context variables and and all observation variables to the fault node observed in the case (i.e., the final diagnosis, as indicated by the diagnostician). This provides a graph skeleton that is subsequently quantified in the following way. All prior probability distributions are set to 0.1/0.9. All conditional probability distributions are set to 0.8/0.2. The EM algorithm, which Marilyn subsequently invokes, treats these initial values as a prior probability distributions and refines them by means of the records accrued during the past diagnostic cases. While the above priors are arbitrary, we found that they are capable of inducing reasonable behavior on the part of the model, even if the number of existing records is small. There is an increasing body of evidence that the precise values of the parameters are not crucial in practice (Pradhan, Henrion, Provan, del Favero, & Huang, 1996; Oniśko & Druzdzel, 2011).

The final model, i.e., model obtained after the parameter refinement stage, is used by Marilyn to generate a list of most likely diagnoses for the current diagnostic case.

4 Empirical Evaluation

4.1 The Data

We tested the accuracy of Marilyn on four different real data sets listed in Table 2. The computing lab data set was collected over the course of two semesters at a help desk of a University of Pittsburgh campus computing lab. Typical campus computing lab help desk problems involve printing problems and printer troubleshooting. Among the four hundred cases in the data set, there are a total of 16 different observations, 12 different context variables, and 21 different problems. The remaining three data sets originate from the UCI Machine Learning repository and were selected based on the following four criteria:

- The data include a known class variable.
- The attribute types of all variables are discrete. We wanted to avoid the need for discretization, which could become a factor confounding our experiment.
- The number of cases in the data file should be over 100, which we believe to be large enough for the purpose of the experiment.
- The data should have been used in the literature in the past, so that we have information about baseline performance of learning algorithms.

The three Irvine repository data sets that fulfilled the above requirements were *SPECT Heart*, *Breast Can-*

cer and *Lymphography*. Their properties are listed in Table 2. #I in the table denotes the number of data records, #A denotes the number of attributes, #CV denotes the number of class variables, and MV describes presence of missing values.

Table 2: Data sets used in our experiments.

Dataset	**#I**	**#A**	**#CV**	**MV**
Computer lab	400	49	21	No
SPECT Heart	267	23	2	No
Breast Cancer	286	10	2	Yes
Lymphography	148	19	4	No

4.2 Methodology

We wanted to test the accuracy of Marilyn as a function of the number of cases that it has seen on each of the data sets listed in Table 2. This is of interest because the idea of query-based diagnostics is meant to work especially when there are no data that can be used to learn a model. Availability of a complete data set would make Marilyn useless, as the model could be learned from data by means of any of the Bayesian network learning methods available in the literature.

We imitated Marilyn's diagnostician's work-flow, which consists of entering three types of information: context information, observations, and the final diagnosis. While, in case of the computing lab help desk data, we had full knowledge of the three types of information, we did not know which of the features in the Irvine medical data sets were context variables and which were observations. Effectively, we treated all features in these data sets as observations. This is a conservative assumption, as it is an additional handicap for Marilyn in the experiments. The effect of our treatment of the medical data was that Marilyn constructed two layer BN2O networks in these cases, similarly to the QMR-DT model (Middleton et al., 1991).

We ran Marilyn 30 times for each data set, randomizing each time the order of records in the data file. The order of the records offered to Marilyn may affect its performance and presenting different orders allows us to observe a range of behaviors. We used the simplest possible criterion in making a diagnostic decision and assumed that the most likely diagnosis is Marilyn's final diagnosis. This is, again, a conservative assumption, as the system displays the top n most likely diagnoses and this gives the user a chance to improve on the system, especially in the early stages, when the model is very crude.

4.3 The Results

4.3.1 Marilyn Results

We calculated Marilyn's cumulative accuracy after each record, so as to know how this performance develops as a function of the number of diagnostic cases that the system has seen. Figure 4 shows the average accuracy of Marilyn as a function of the number of cases for each of the four data sets with range of the curves (vertical bars) plotted for selected number of records. The plots show that while Marilyn was rather weak in the beginning (during the first thirty cases or so), it became quite accurate after roughly 70 to 100 cases (this varied per data set). Interestingly, in case of the SPECT data set, Marilyn reached the accuracy of over 60% after fewer than ten cases. In all data sets, 40 or so cases were sufficient to reach a reasonable accuracy. This accuracy not only improved over time but also improved reliably, as indicated by smaller variance in the results of different random orders of records. Interestingly, there is some similarity between the plots of Marilyn's performance, as in Figure 4, and the so called *power curve of practice* in the psychology literature (Newell & Rosenbloom, 1981).

4.3.2 Marilyn's Relative Performance

Cumulative accuracy for the last record entered is the final accuracy result of Marilyn on the data set. Marilyn's final performance on the four data sets was 90.25%, 78.75%, 77.18%, and 69.95% for the *Computer Lab*, *SPECT Heart*, *Breast Cancer* and *Lymphography* data respectively (see the extreme right cumulative performance in Figure 4). It has to be added that in achieving this result Marilyn has seen (i.e., was trained on) the average of 50% of the records. When processing the first case, Marilyn has seen zero prior cases, when processing the 10th case, it has used only 9 preceding cases, when processing the last, nth case, it has seen $n-1$ preceding cases. The average number of training records is thus $n/2$.

Table 3: Accuracy comparison results with Bayesian approaches using leave-one-out cross validation

Dataset	Marilyn	**Naive Bayes**	**GTT**
CompLab	94.50%	94.25%	91.25%
SPECT	79.40%	79.40%	78.65%
BC	68.18%	42.57%	47.97%
Lymph	81.08%	66.08%	67.83%

In order to disambiguate the specific procedure that we used to obtain Marilyn's cumulative performance from the capability of the learning function by it-

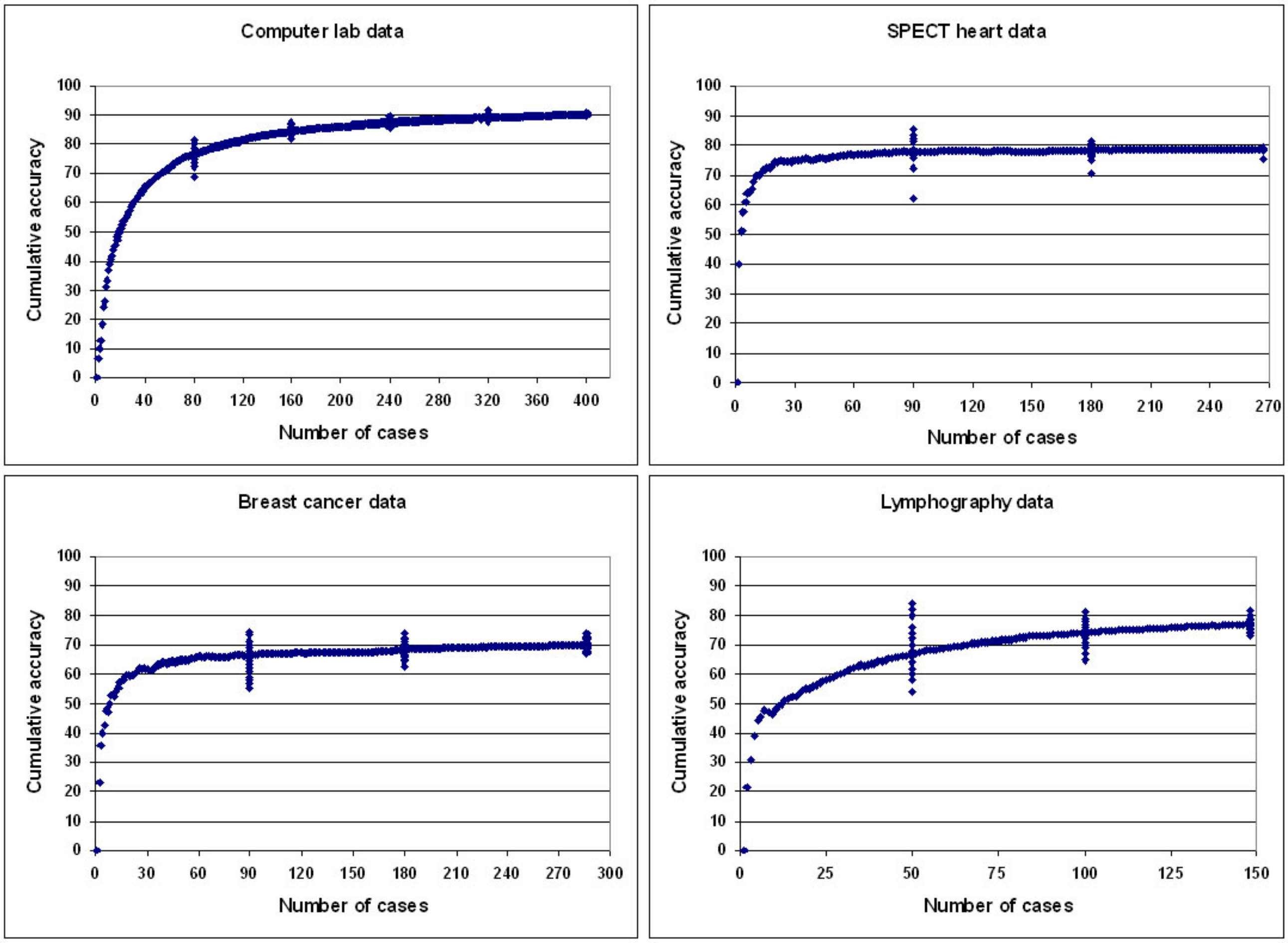


Figure 4: Marilyn's cumulative accuracy as a function of the number of cases seen

self, we performed an experiment in which we allowed Marilyn to learn from all available records alongside with two Bayesian learning algorithms: (1) Naive Bayes (Langley, Iba, & Thompson, 1992), and (2) a Bayesian search algorithm Greedy Thick Thinning (GTT) (Dash & Druzdzel, 2003). We used the leave-one-out cross validation to measure the accuracy of the three classifiers, assuming that the diagnosis is correct when the most probable class matches the correct class. We show the results of this experiment in Table 3. Marilyn performed better than Naive Bayes and GTT on all data sets. We believe that some of Marilyn's power comes from its priors and structural information extracted from the data.

The three data sets that we chose for our experiments have been subject of experiments published in the literature. The best accuracy result for SPECT heart data with CLIP3 machine learning algorithm is 84% (Kurgan, Cios, Tadeusiewicz, Ogiela, & Goodenday, 2001). The best accuracy achieved on the Breast cancer data was by means of k-nearest neighbor (k-NN) algorithm and amounted to 79.5% (Kononenko, Bratko, & Kukar, 1997). The best accuracy on the Lymphography set was achieved by means of the Tree-Augmented Naive Bayes algorithm and was 85.47% (Madden, 2002). We compared Marilyn's performance to each of these, repeating the experiment under the same conditions, i.e., with precisely the same cross-validation method as used in the experiments reported in the literature. Table 4 shows the accuracy for each of the data sets and each of the algorithms.

Table 4: Accuracy comparison results with state of the art approaches

Dataset	Marilyn	**CLIP3**	**k-NN**	**TAN**
SPECT	93.58%	84%	N/A	N/A
BC	73.02%	N/A	79.50%	N/A
Lymph	81.92%	N/A	82.60%	85.47%

While Marilyn's accuracy is typically lower than that of the state of the art learning algorithms, it is certainly in the same ballpark. We would like to point out that the best results reported in the literature belong to different algorithms, i.e., there seems to be no algorithm that is uniformly best on all data sets. If the same algorithm were applied to all four data sets, there is a good chance that its accuracy on some of these could be worse than the accuracy of Marilyn.

5 Conclusion

Query-based diagnostic offers passive, incremental construction of diagnostic models based on the interaction between a diagnostician and a computer-based diagnostic system. Effectively, this approach eliminates knowledge engineering, the main bottleneck in practical application of Bayesian networks.

While this idea is appealing, it has undergone only limited testing in practice. In this paper, we described a series of experiments that subject a prototype implementing passive, incremental model construction to a rigorous practical test. Data obtained from the Irvine repository made the evaluation fairly realistic. The results of our experiments show that a system like Marilyn is capable of giving reasonable suggestions after a modest number of observed cases. Performance in the order of 70-90% typically occurred not later than after roughly 40 cases. Even though this experiment offers just a few data points and this type of systems need to be tested more in practice, we believe that the result is very promising and compares favorably with state of the art approaches based on learning.

Acknowledgements

This work has been supported by the National Institute of Health under grant number U01HL101066-01, the Government of the Kingdom of Thailand, the Air Force Office of Scientific Research grant FA9550–06–1–0243, and by Intel Research. Implementation of Marilyn is based on SMILE, a Bayesian inference engine developed at the Decision Systems Laboratory and available at `http://genie.sis.pitt.edu/`. We would like to express our thanks to the DSL members for their valuable suggestions. Special thanks go to Erik Pols, Mark Voortman and Adam Zagorecki.

References

Abbasi, A., Dailey, M., Afzulpurkar, N., & Uno, T. (2010, March). Student mental state inference from unintentional body gestures using dynamic Bayesian networks. *Journal on Multimodal User Interfaces*, *3*(1), 21–31.

Agosta, J. M., Gardos, T. R., & Druzdzel, M. J. (2008). Query-based diagnostics. In M. Jaeger & T. D. Nielsen (Eds.), *Proceedings of the Fourth European Workshop on Probabilistic Graphical Models (PGM–08)* (pp. 1–8). Aalborg, Denmark.

Agosta, J. M., Khan, O. Z., & Poupart, P. (2010). Evaluation results for a query-based diagnostics application. In P. Myllymaki, T. Roos, & T. Jaakkola (Eds.), *Proceedings of the Fifth European Workshop on Probabilistic Graphical Models (PGM–10)* (pp. 1–9). Helsinki, Finland.

Breese, J. S., Goldman, R. P., & Wellman, M. P. (1994). Introduction to the special section on knowledge-based construction of probabilistic and decision models. *IEEE Transactions on Systems, Man and Cybernetics*, *24*(11), 1577–1579.

Cooper, G. F., & Herskovits, E. (1992). A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, *9*(4), 309–347.

Dash, D., & Druzdzel, M. J. (2003). Robust independence testing for constraint-based learning of causal structure. In C. Meek & U. Kjærulff (Eds.), *UAI* (pp. 167–174). Morgan Kaufmann.

Dempster, A., Laird, N., & Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, *39*(1), 1–38.

Díez, F. J. (1993). Parameter adjustment in Bayes networks. The generalized Noisy-OR gate. In *Proceedings of the Ninth Annual Conference on Uncertainty in Artificial Intelligence (UAI–93)* (pp. 99–105). Washington, D.C..

Díez, F. J., & Druzdzel, M. J. (2006). *Canonical probabilistic models for knowledge engineering* (Tech. Rep.). UNED, Madrid, Spain. CISIAD-06-01.

Heckerman, D. (1990, August). Probabilistic similarity networks. *Networks*, *20*(5), 607–636.

Henrion, M. (1989). Some practical issues in constructing belief networks. In L. Kanal, T. Levitt, & J. Lemmer (Eds.), *Uncertainty in Artificial Intelligence 3* (pp. 161–173). New York, N. Y.: Elsevier Science Publishing Company, Inc.

Ibargengoytia, P. H., Vadera, S., & Sucar, L. E. (2006). A probabilistic model for information and sensor validation. *The Computer Journal*, *49*, 113–126.

Koller, D., & Pfeffer, A. (1997). Object-oriented Bayesian networks. In *Proceedings of the Thirteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–97)* (pp. 302–313). San Francisco, CA: Morgan Kaufmann Publishers.

Kononenko, I., Bratko, I., & Kukar, M. (1997). Application of machine learning to medical diagnosis. In I. Michalski R.S. Bratko & M. Kubat (Eds.), *Machine Learning, Data Mining and Knowledge Discovery: Methods and Applications.* John Wiley & Sons.

Kraaijeveld, P., & Druzdzel, M. J. (2005, June 1–3). *GeNIeRate*: An interactive generator of diagnostic Bayesian network models. In *Working Notes of the 16th International Workshop on Principles of Diagnosis (DX–05)* (pp. 175–180). Monterey, CA.

Kurgan, L. A., Cios, K. J., Tadeusiewicz, R., Ogiela, M. R., & Goodenday, L. S. (2001). Knowledge discovery approach to automated cardiac SPECT diagnosis. *Artificial Intelligence in Medicine*, *23*, 149.

Langley, P., Iba, W., & Thompson, K. (1992). An analysis of Bayesian classifiers. In *Proceedings of the Tenth National Conference on Artificial Intelligence (AAAI–92)* (pp. 223–228). MIT Press.

Laskey, K. B., & Mahoney, S. M. (1997). Network fragments: Representing knowledge for constructing probabilistic models. In *Proceedings of the Thirteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–97)* (pp. 334–341). San Francisco, CA: Morgan Kaufmann Publishers.

Lu, T.-C., Druzdzel, M. J., & Leong, T.-Y. (2000). Causal mechanism-based model construction. In *Uncertainty in Artificial Intelligence: Proceedings of the Sixteenth Conference (UAI–2000)* (pp. 353–362). San Francisco, CA: Morgan Kaufmann Publishers.

Madden, M. G. (2002). Evaluation of the performance of the Markov blanket Bayesian classifier algorithm. *CoRR*, *cs.LG/0211003*.

Mahoney, S. M., & Laskey, K. B. (1996). Network engineering for complex belief networks. In *Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence (UAI–96)* (pp. 389–396). San Francisco, CA: Morgan Kaufmann Publishers.

Middleton, B., Shwe, M., Heckerman, D., Henrion, M., Horvitz, E., Lehmann, H., et al. (1991). Probabilistic diagnosis using a reformulation of the INTERNIST–1/QMR knowledge base: II. Evaluation of diagnostic performance. *Methods of Information in Medicine*, *30*(4), 256–267.

Natarajan, S., Tadepalli, P., Altendorf, E., Dietterich, T. G., Fern, A., & Restificar, A. (2005). Learning first-order probabilistic models with combining rules. In *Proceedings of the 22nd International Conference on Machine Learning* (pp. 609–616). New York, NY, USA: ACM.

Newell, A., & Rosenbloom, P. S. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive Skill and Their Acquisition* (pp. 1–55). Hillsdale, NJ: Lawrence Erlbaum Associates.

Oniśko, A., & Druzdzel, M. J. (2011). Impact of quality of Bayesian network parameters on accuracy of medical diagnostic systems. In *Working Notes of the 2011 AIME'11 Workshop on Probabilistic Problem Solving in Biomedicine.* Bled, Slovenia.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference.* San Mateo, CA: Morgan Kaufmann Publishers, Inc.

Pearl, J., & Verma, T. S. (1991). A theory of inferred causation. In J. Allen, R. Fikes, & E. Sandewall (Eds.), *Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference (KR–91)* (pp. 441–452). Cambridge, MA: Morgan Kaufmann Publishers, Inc., San Mateo, CA.

Pfeffer, A., Koller, D., Milch, B., & Takusagawa, K. T. (1999). SPOOK: A system for probabilistic object-oriented knowledge representation. In *Proceedings of the Fifteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–99)* (pp. 541–550). San Francisco, CA: Morgan Kaufmann Publishers.

Pols, E. (2007, April). *Marilyn: A guided maintenance system that represents direct probabilistic influences among diagnostic knowledge* (Technical Report). Delft, The Netherlands: Delft University of Technology.

Pradhan, M., Henrion, M., Provan, G., del Favero, B., & Huang, K. (1996, August). The sensitivity of belief networks to imprecise probabilities: An experimental investigation. *Artificial Intelligence*, *85*(1–2), 363–397.

Ratnapinda, P., & Druzdzel, M. J. (2009). Passive construction of diagnostic decision models: An empirical evaluation. In *International Multiconference on Computer Science and Information Technology (IMCSIT–09)* (pp. 601–607).

Spirtes, P., Glymour, C., & Scheines, R. (1993). *Causation, prediction, and search.* New York: Springer Verlag.

Srinivas, S. (1993). A generalization of the noisy-OR model. In *Proceedings of the Ninth Annual Conference on Uncertainty in Artificial Intelligence (UAI–93)* (pp. 208–215). San Francisco, CA: Morgan Kaufmann Publishers.

Vomlel, J. (2006, March). Noisy-OR classifier: Research articles. *International Journal of Intelligent Systems*, *21*, 381–398.