A USER-FRIENDLY DEVELOPMENT TOOL FOR MEDICAL DIAGNOSIS BASED ON BAYESIAN NETWORKS

Isabel Milho*‡, Ana Fred‡**

* Instituto Superior de Engenharia de Lisboa, DEEC,
R. Conselheiro Emídio Navarro, 1, 1900 Lisboa, Portugal, e-mail: imilho@isel.pt

** Instituto Superior Técnico, Av. Rovisco Pais, 1049-001 Lisboa, Portugal

‡ Instituto de Telecomunicações - Pólo de Lisboa, Torre Norte, Piso 10,
Av. Rovisco Pais, 1049-001 Lisboa, Portugal, e-mail: eanafred@beta.ist.utl.pt

Abstract: This paper presents a web supported development tool specific for medical diagnosis, based on Bayesian networks. The proposed system provides a user-friendly interface, giving the users (experts in the medical domain) the possibility to design diagnostic applications without deep background knowledge on Bayesian networks theory. Efficient statistical inference mechanisms are implemented, taking advantage of the simple structure of medical diagnostic models, composed of causal disease-symptom relations. The developed system also provides an Internet page interface to interact with the expert, when using the designed application for diagnosis. Illustrating the functionality of the proposed tool, an application in the context of sleep-disorders diagnosis is presented.

Keywords: Bayesian networks, diagnostic development tools, decision support systems on the web.

1. Introduction

Computer assisted techniques have been proposed in the past to help solving the diagnostic problem [Giarratano94], [Russel95]. Commonly known as *expert systems*, distinct approaches for modelling domain knowledge include: rule-based or production systems, frame-based systems, semantic networks, neural networks, and Bayesian networks (BNs). Explanatory mechanisms are more easily implemented with rule-based systems and graphical models, such as the BN paradigm, making these preferred solutions to the medical diagnostic problem.

Bayesian networks [Pearl88] [Russel95] [Jensen96] [Frey98] are a powerful and sound formalism that allows reasoning under uncertainty, proposing a graphical representation of statistical dependencies between domain variables. Simultaneously to the application to a wide range of domains, recent developments in this area include new and more efficient inference methods, as well as universal tools for the design of BN-based applications [Hugin99] [MSBN99] [JavaBayes99]. An extended list of software for BNs can be found in [BNS99]. These tools range from simple software modules to versatile, graphical interfaces that assist the designer of new applications. Offering multiple options in an application independent framework, the use of such tools often requires a deep knowledge of the BN's formalism and methods.

In this paper we propose a BN-based development tool, specific for medical diagnosis on the web. The main feature of the proposed system is to provide a simple and integrated tool for designing diagnostic applications, not requiring deep knowledge on BNs background theory. Moreover, the proposed tool produces the diagnosis results in an output layout specially designed for the user-expert analysis. In addition, the user interface (when diagnosing) is made through an Internet browser. Finally, considering that most of the medical diagnostic models have a simple structure (composed of causal disease-symptom relations), the proposed tool implements efficient statistical inference mechanisms,

taking advantage of that simplified model, thus relieving the computational burden.

The diagnostic model based on Bayesian networks is introduced in section 2. The diagnostic development tool is described in section 3; details about system architecture are given in section 4. An example, concerning the application of the system in the context of sleep-disorders diagnosis is presented in section 5. Finally, conclusions and suggestions for further work are presented in section 6.

2. BN-Based Diagnostic Model

The problem of medical diagnosis can be stated as follows: given a set of *symptoms* (clinical data) and *signals,* or *test results* (tests performed on the patient), assess pathological situations identifying which *diseases* justify the particular findings.

Bayesian networks provide a probabilistic model for uncertain reasoning, adequate for handling the diagnostic problem. A BN for a given domain represents the joint probability distribution, $p(\mathbf{x})$, over the set of variables, $\mathbf{X}$, of the domain, as a set of local distributions combined with a set of conditional independence assertions; that allows the construction of the global joint probability distribution from the local distributions as

$$p(\mathbf{x}) = \prod_{i=1}^{n} p(x_i \mid pa(x_i)) \qquad (1)$$

where, for each variable x_i, $pa(x_i) \subseteq \{x_1, \ldots, x_{i-1}\}$ is a set of variables that renders x_i and $\{x_1, \ldots, x_{i-1}\}$ conditionally independent. The BN structure encodes the assertions of conditional independence as a directed acyclic graph such that: (a)- each node corresponds to a variable; (b)- the parents of the node corresponding to x_i are the nodes associated to the variables in $pa(x_i)$. The pair formed by the structure (graph) and the collection of local distributions, $p(x_i \mid pa(x_i))$, for each node in the domain, constitute the Bayesian network for that domain.

Concerning the context of medical diagnosis, a Bayesian network is a directed acyclic graph in which diseases and symptoms, as well as tests results, are graphically represented as nodes (see Fig. 1), arcs expressing causal relationships between these variables.
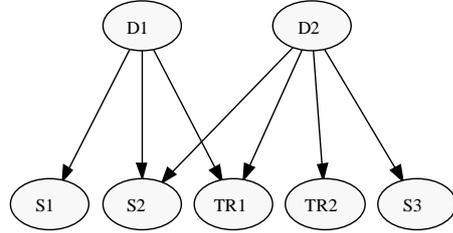
Figure 1. Example of a Bayesian network modelling the classical diagnostic problem. The root nodes (*D*-nodes) represent diseases, while the descendants model symptoms and tests results (*S*-nodes and *TR*-nodes respectively). A *D*-node connects to an *S*-node if the first is a possible cause for the symptom in the *S*-node.

Because the model is built in causal form (arrows in the network point from causes to effects), designing the graphical representation of a diagnostic problem is reasonably straightforward, using medical knowledge of the domain.

The basic task for the diagnostic system, under a probabilistic framework, is to compute the posterior probability distribution for a set of query variables, given exact values for some evidence variables. That is, the system computes the probability P(*Query|Evidence*). In the example of Fig. 1, *D*-node D_1 is a possible query variable and S_1 and TR_1 could serve as evidence variables – typical diagnostic reasoning. In Bayesian networks any node can serve as either a query or an evidence variable. Four distinct kinds of inference can be performed:

- Diagnostic inference (from effects to causes). Ex: given a symptom S_1 infer the probability of the pathology D_1, P($D_1|S_1$).
- Causal inference (from causes to effects). Ex: given disease D_2 find the most likely symptoms, P($S_i|D_2$).
- Inter-causal inference (between causes of a common effect). Ex: given S_2 infer P($D_1| S_2$), but adding evidence that D_2 is true makes the probability of D_1 go down. Although D_1 and D_2 nodes are

independent the presence of one makes the other less likely.

- Mixed inferences (combining two or more of the above).

3. Development-Tool Functionality

The proposed system provides an integrated development tool for designing specific diagnostic applications. With an easy-to-use interface, this tool gives the user the possibility to design his particular application, not requiring knowledge of BNs background theory.

The development tool essentially provides a graphical interface for domain knowledge introduction, and a Web page (input/output) interface to interact with the system for diagnosing.

Two interaction phases are thus supported: (1)- *design* of an application in a particular domain; (2)- routine *usage* of the application as an auxiliary diagnostic system.
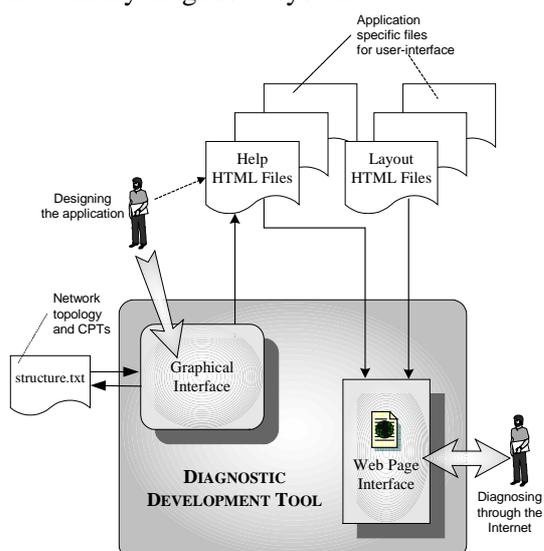
Figure 2. System interface and functionality.

In the first phase, a graphical interface (see Fig. 2) enables a structured introduction of the domain knowledge, ultimately defining the topology and local distributions of the BN. The main concepts/components handled by the graphics interface are the diseases, symptoms and test results (corresponding to *D*-nodes and *S*-nodes as well as *TR*-nodes referred previously). Despite of its default,

boolean-valued (with *absent* and *present* values), the user can choose each variable (node) to be multiple-valued (for example, to characterise a test result with possible values: *low*, *normal*, *high* and *too high*). At the same time, the user is allowed to add help text to each node, that latter on will be accessed via Internet.

Additionally to designing the network (knowledge in the given medical context), the user has the possibility to customise help information by simply writing HTML *Help* files. Moreover, layout customisation of the diagnostic interface is also possible through changes in the respective HTML *Layout* files.

A preliminary version of this system used HUGIN environment [Hugin99], [Jensen96] to do the graphical interface (taking advantage of the HUGIN output file describing the user-defined structure). The complete graphical interface is currently under development.

After the application has been designed, the expert accesses the system through an Internet browser – *usage* phase. By interacting with the Internet page, the user can operate for diagnosing, putting in action the diagnostic system with its inference mechanism. Fig. 5 shows an example of the interface, corresponding to the application described in section 5. Any variable node can be set as evidence, all typical queries described in section 2 being possible through this interface (diagnostic, causal, inter-causal and mixed inferences).

4. System Architecture

A block diagram representing the architecture and implementation of the diagnostic development tool is presented in Fig. 3.

The system consists of three major parts: (a) the inference algorithm module; (b) a Java interface module; (c) the diagnostic system input/output - Internet page.

The inference algorithm is implemented in C/C++ language and its main function is the computation of the probabilities of query-variables: given the evidence nodes (entered by the user in the Internet page and

stored in *evidences.txt*) computes the probabilities for the other nodes. The output file is produced with all actual variable probabilities (*probabilities.txt*). Considering that most of the medical diagnostic models have a simple structure (composed of causal disease-symptom relations), an efficient algorithm was developed based on a forward-backward procedure [Pearl88], [Russel95], [Frey98].
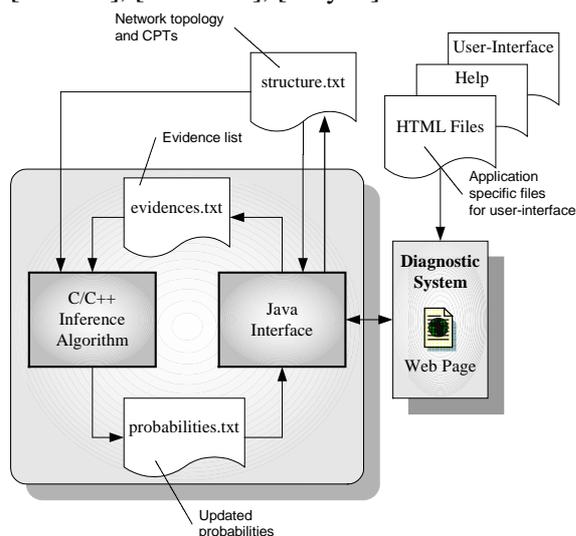
Figure 3. Block diagram representing the diagnostic system architecture and implementation.

The Java interface module supports the information exchange between the Internet diagnostic-system page and the inference algorithm. Hiding particular aspects of the implementation, this module gives the users a friendly interface. The graphical tool currently being developed is also integrated in the Java interface module. Notice (in Fig. 3) that this module outputs information to the external *structure.txt* file.

Forming the core of the system, these two modules interact with the diagnostic-system Internet page; this provides the interface with the user for input/output. Its associated HTML files provide the designed output layout and help files.

When designing the application the user indirectly introduces knowledge in *structure.txt* and HTML help files. Later, when opening the diagnostic-system Internet page, the network structure and local distributions are loaded to the inference module, reading data from the

structure.txt file. After requested by the user, the diagnostic system runs the inference algorithm. The Internet page exchanges data with the Java interface (evidences and updated probabilities) each time the user interacts with the system.

5. Sleep-Disorders Diagnosis

An example of an application of this tool is the *Sleep-Disorders Diagnostic System* (SDDS). It is a medical decision-support system to help in diagnosing sleep disorders, using clinical and polysomnographic data. The development of this application was based on previous work on the topic [Filipe99].

Part of the domain knowledge is represented in Fig. 4, showing the network structure concerning the following pathologies: *Psycophysiological Insomnia*, *Idiopathic Insomnia*, *Obstructive Sleep Apnea Syndrome* and *Narcolepsy*; the corresponding *D*-nodes are respectively PSIC, IDIO, OSAS and Narco. The network topology and its conditional probabilities were specified from medical literature [Thorphy97] and from consultation with experts in the field.
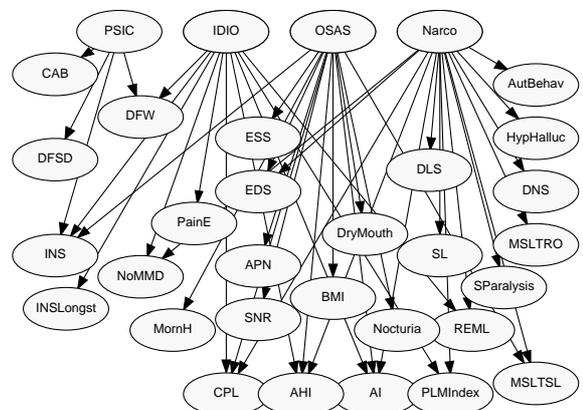
Figure 4. Bayesian network topology modelling the sleep-disorders diagnostic problem.

The user accesses the SDDS via an Internet browser, which displays an output like the one illustrated in Fig. 5. The user interface also provides an on-line manual (supported by HTML help files) with the explanations for all the pathologies and symptoms, as well as the polysomnographic data.
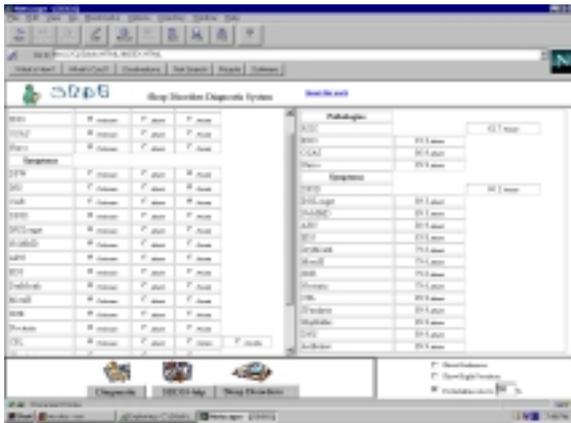
Figure 5. The SDDS Internet page, showing the results of a diagnosis.

6. Conclusions

A development tool for designing medical diagnosis applications was presented. With a user-friendly interface, this tool offers the experts the possibility to design their particular applications without BNs background knowledge. Taking advantage of the simple structure of diagnostic models (composed of causal disease-symptom relations), an efficient statistical inference mechanism is implemented, relieving the computational burden. Moreover the developed system provides an Internet page based interface when using the designed application for diagnosis.

An application of this tool in the context of sleep-disorders diagnosis (SDDS) was presented, illustrating the system functionality.

Future work includes the embedding of learning algorithms in the software tool [Heckerman96], validation and comparative studies of the SDDS [Filipe99], as well as the application of the tool to other medical domains.

Moving towards a *Bayesian Network Interchange Format* (BNIF) represented in XML[1], work is already undertaken adopting the XBN format [XBN99] proposed by the *Decision Theory and Adaptive Systems Group* at *Microsoft Research*. Adding this facility to the system-interface presents new advantages: the possibility to use MSBN free software,

increased versatility and future compatibility of the system with similar tools.

7. Acknowledgements

8. References

[Thorphy97] Diagnostic Classification Steering Committee, Thorpy M. J. (Chairman), 1997, *International Classification of Sleep Disorders: Diagnostic and Coding Manual (Revised)*, Rochester, Minnesota: American Sleep Disorders Association.

[Filipe99] Filipe, J., Fred, A., Fernandes, M., 1999, GDOS - A Graphical Diagnostic-Oriented Expert System Development Tool, *Proceedings of the 1st International Conference on Enterprise Information Systems*, ICEIS'99, 211-218.

[Frey98] Frey, B. J., 1998, *Graphical Models for Machine Learning and Digital Communication*, The MIT Press.

[Giarratano94] Giarratano, J., Riley, G., 1994, *Expert Systems: Principles and Programming*, Boston: PWS Publishing Company.

[Heckerman96] Heckerman, D., revised in 1996, A Tutorial on Learning with Bayesian Networks, *Microsoft Research Technical Report MSR-TR--95-06*.

[Hugin99] HUGIN Expert A/S, in http://www.hugin.dk

[JavaBayes99] JavaBayes - Bayesian Networks in Java, in http://www.cs.cmu.edu/~javabayes

[Jensen96] Jensen, F., 1996, *An Introduction to Bayesian Networks*, Springer-Verlag New York, Inc.

[MSBN99] Microsoft Belief Network Tools, in http://www.research.microsoft.com/dtg/msbn/

[Pearl88] Pearl, J., 1988, *Probabilistic Reasoning in Intelligent Systems: networks of plausible inference*, Morgan Kaufmann Publishers, Inc.

[Russel95] Russel, S., Norvig, P., 1995, *Artificial Intelligence: A Modern Approach*, Prentice-Hall, Inc.

[BNS99] List of Belief Networks Software, in http://bayes.stat.washington.edu/almond/belief.html

[XBN99] XBN, a XML Belief Network File Format, in http://www.research.microsoft.com/dtas/bnformat/

---

[1] The *Extensible Markup Language*, representation introduced by the World Wide Web Consortium