BAYESIAN NETWORKS: A MODEL OF SELF-ACTIVATED MEMORY FOR EVIDENTIAL REASONING

Judea Pearl

June 1985
CSD-850021

BAYESIAN NETWORKS: A MODEL OF SELF-ACTIVATED MEMORY FOR EVIDENTIAL REASONING*

Judea Pearl
Cognitive Systems Laboratory, Computer Science Department, UCLA

ABSTRACT

The paper reports recent results from the theory of Bayesian networks, which offer a viable formalism for realizing the computational objectives of connectionist models of knowledge. In particular, we show that the Bayesian network formalism is supportive of self-activated, multidirectional propagation of evidence that converges rapidly to a globally-consistent equilibrium.

1. INTRODUCTION

This study was motivated by attempts to devise a computational model for humans' inferential reasoning, namely, the mechanism by which people integrate data from various sources and generate a coherent interpretation of that data. Since the knowledge from which inferences are drawn is mostly judgmental--namely, subjective, uncertain, and incomplete--a natural place to start would be to cast the reasoning process in the framework of probability theory. Probability theory is also useful because it is the simplist calculus which permits inferences to flow two ways: from hypothesis to evidence (predictive), as well as from evidence to hypothesis (diagnostic). Unfortunately, traditional probability theory has erected cultural barriers against its usage in modelling human cognition. Scholarly textbooks on probability theory try hard to create the impression that to construct an adequate representation of probabilistic knowledge we must first define a ***joint distribution function*** on all propositions and their combinations, and that this function should serve as the basis for all inferred judgements– a rather distorted picture of human reasoning.

Human judgments regarding a small number of propositions (such as the likelihood that a patient suffering from a given disease will develop a certain type of complication) are issued swiftly and reliably, while judging the likelihood of a conjunction of many propositions is done with great degree of difficulty and hesitancy. This suggests that the elementary building blocks which make up human knowledge are not entries of a giant joint-distribution table, but rather low-order probabilistic relations between small clusters of semantically-related propositions.

Additionally, a person reluctant to giving a numerical estimate for the conditional probability $P(A|B)$, will normally show no hesitation to state whether propositions A and B are dependent or independent,given C, namely, whether knowing the truth of B will or will not alter the belief in A, assuming that C is true. Evidently, the notion of conditional dependence is more basic than the numerical values attached to probability judgments, contrary to the picture painted in most textbooks on probability theory, where the latter is presumed to provide the criterion for testing the former. This suggests that the fundamental structure of human judgmental knowledge can be represented by ***dependency graphs*** and that mental tracing of links in these graphs are responsible for the basic steps in querying and updating that knowledge. Bayesian networks offer an effective formalism for these graph operations.

*This work was supported in part by the National Science Foundation, Grant #DSR 83-13875

2. BAYESIAN NETWORKS

Bayes Networks are directed acyclic graphs in which the nodes represent propositions (or variables), the arcs signify the existence of direct causal influences between the linked propositions, and the strengths of these influences are quantified by conditional probabilities (Figure 1).

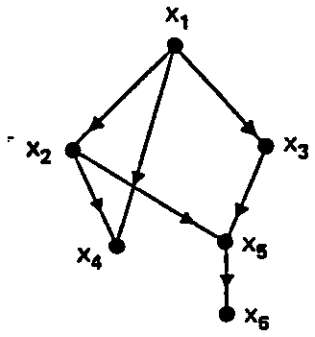

Figure 1

Thus, if the graph contains the variables $x_1, \ldots, x_n$, and S_i is the set of parents for variable x_i, then a complete and consistent quantification can be attained by specifying, for each node x_i, an assessment $P'(x_i \mid S_i)$ of $P(x_i \mid S_i)$. The product of all these assessments,

$$P(x_1, \ldots, x_n) = \prod_i P'(x_i|S_i) \tag{1}$$

constitutes a joint-probability model which supports the assessed quantities. That is, if we compute the conditional probabilities $P(x_i \mid S_i)$ dictated by $P(x_1, \ldots, x_n)$, the original assessments are recovered. Thus, for example, the distribution corresponding to the graph of Figure 1 can be written by inspection:

$$P(x_1,x_2,x_3,x_4,x_5,x_6) = P(x_6|x_5)\, P(x_5|x_2,x_3)\, P(x_4|x_1,x_2)\, P(x_3|x_1)\, P(x_2|x_1)\, P(x_1).$$

An important feature of Bayes network is that it provides a clear graphical representation for many independence relationships embedded in the underlying probabilistic model. The criterion for detecting these independencies is based on ***graph separation***: namely, if all paths between x_i and x_j are "blocked" by a a subset S of variables, then x_i is independent of x_j given the values of the variables in S. Thus, each variable x_i is independent of both its siblings and its grandparents, given the values of the variables in its parent set S_i. For this "blocking" criterion to hold in general, we must provide a special interpretation of separation for nodes that share common children. We say that the pathway along arrows meeting head-to-head at node x_k is normally "blocked", unless x_k or any of its descendants is in S. In Figure 1, for example, x_2 and x_3 are independent given $S_1 = \{x_1\}$ or $S_2 = \{x_1,x_4\}$, because the two paths between x_2 and x_3 are blocked by either one of these sets. However, x_2 and x_3 may not be independent given $S_3 = \{x_1,x_6\}$, because x_6, as a descendant of x_5 , "unblocks" the head-to-head connection at x_5, thus opening a pathway between x_2 and x_3.

3. AUTONOMOUS PROPAGATION AS A COMPUTATIONAL PARADIGM

Once Bayesian network is constructed, it can be used to represent the generic causal knowledge of a given domain, and can be consulted to reason about the interpretation of specific input data. The interpretation process involves instantiating a set of variables corresponding to the input data and calculating its impact on the probabilities of a set of variables designated as hypotheses. In principle, this process can be executed by an external interpreter who may have access to all parts of the network, may use its own computational facilities, and may schedule its computational steps so as to take full advantage of the network topology with respect to the incoming data. However, the use of such an interpreter seems foreign to the reasoning process normally exhibited by humans [Shastri and Feldman, 1984]. Our limited short-term memory and narrow focus of attention, combined with our inflexibility of shifting rapidly between alternative lines of reasoning seem to suggest that our reasoning process is fairly local, progressing incrementally along prescribed pathways. Moreover, the speed and ease with which we perform some of the low level interpretive functions, such as recognizing scenes, comprehending text, and even understanding stories, strongly suggest that these processes involve a significant amount of parallelism, and that most of the processing is done at the ***knowledge level itself***, not external to it.

A paradigm for modeling such active knowledge base would be to view a Bayesian network not merely as a passive parsimonious code for storing factual knowledge but also as a computational architecture for reasoning about that knowledge. That means that the links in the network should be treated as the only pathways and activation centers that direct and propel the flow of data in the process of querying and updating beliefs. Accordingly, we assume that each node in the network is designated a separate processor which both maintains the parameters of belief for the host variable and manages the communication links to and from the set of neighboring, logically related, variables. The communication lines are assumed to be open at all times, i.e., each processor may at any time interrogate the belief parameters associated with its neighbors and compare them to its own parameters. If the compared quantities satisfy some local constraints, no activity takes place. However, if any of these constraints is violated, the responsible node is activated to revise its violating parameter and set it straight. This, of course, will activate similar revisions at the neighboring nodes and will set up a multidirectional propagation process, until equilibrium is reached.

While constraint-propagation mechanisms have found several applications in AI, such as vision [Rosenfeld, Hummel and Zucker, 1976; Waltz, 1972] and truth maintenance [McAllester, 1980], their use in evidential reasoning has been limited to non-Bayesian formalisms [e.g. Lowrance, 1982, Shastri and Feldman, 1984]. The reason has been several-fold.

First,the conditional probabilities characterizing the links in the network do not seem to impose definitive constraints on the probabilities that can be assigned to the nodes. The quantifier $P(A|B)$ only restricts the belief accorded to A in a very special set of circumstances: namely, when B is known to be true with absolute certainty, and when no other evidential data is available. Under normal circumstances, all internal nodes in the network will be subject to some uncertainty and, more seriously, after observing evidence e the conditional belief in A is no longer governed by $P(A|B)$ but by $P(A|B, e)$, which may be totally different. The result is that any assignment of beliefs, $P(A)$ and $P(B)$, to propositions A and B can be consistent with the value of $P(A|B)$ initially assigned to the link connecting them; therefore, no violation of constraint can be detected locally.

Next, the difference between $P(A|B, e)$ and $P(A|B)$ seems to suggest that the weights on the links should not remain fixed but should undergo constant adjustment as new evidence ar-

rives. This, in turn, would require an enormous computational work and would wipe out the advantages normally associated with propagation through fixed constraints.

Finally, the fact that evidential reasoning involves both top-down (predictive) and bottom-up (diagnostic) inferences has caused apprehensions that, once we allow the propagation process to run its course unsupervised, pathological cases of instability, deadlock, and circular reasoning will develop [Lowrance, 1982]. Indeed, if a stronger belief in a given hypothesis means a greater expectation for the occurrence of its various manifestations and if, in turn, a greater certainty in the occurrence of these manifestations adds further credence to the hypothesis, how can one avoid infinite updating loops when the processors responsible for these propositions begin to communicate with one another?

This paper reports that coherent and stable probabilistic reasoning *can* be accomplished by local propagation mechanisms while keeping the weights on the links constant throughout the process. This is made possible by characterizing the belief in each proposition by a *vector* of several parameters, each representing the degree of support that the host proposition obtains from one of its neighbors. Maintaining such a breakdown record of the sources of belief is also postulated as the mechanism which permits people to trace back reasoned assumptions for the purposes of modifying the model and generating explanatory arguments.

4. PROPAGATION IN SINGLY-CONNECTED NETWORKS

The problems associated with asynchronous propagation of beliefs, can be solved completely if the network is singly connected, namely, if there is one underlying path between any pair of nodes. These include trees, where each node has a single parent, as well as graphs with multi-parent nodes, representing events with several causal factors. The analysis of trees is carried out in Pearl [1982], and the extension to general singly connected graphs is reported in Kim and Pearl [1983]. In both cases, the belief-updating scheme possesses the following properties:

1. New information diffuses through the network in a single pass, i.e., equilibrium is reached in time proportional to the diameter of the network.

2. The primitive processors are simple, repetitive, and they require no working memory except that used in matrix multiplication.

3. The local computations and the final belief distribution are entirely independent of the control mechanism that activates the individual operations. They can be activated by either data-driven or goal-driven (e.g., requests for evidence) control strategies, by a clock, or at random.

Thus, this architecture lends itself naturally to hardware implementation, capable of real-time interpretation of rapidly changing data. It also provides a reasonable model of neural nets involved in cognitive tasks such as visual recognition, reading comprehension [Rumelhart, 1976], and associative retrieval [Anderson, 1983], where unsupervised parallelism is an uncontested mechanism.

5. MANAGING LOOPS AND THE DEVELOPMENT OF CAUSAL MODELS

The efficacy of singly-connected networks in supporting autonomous propagation raises the question of whether similar propagation mechanisms can operate in less restrictive networks (like the one in Figure 1), where multiple parents of common children also possess common ancestors, thus forming loops in the underlying network. If we ignore the existence of loops and permit the nodes to continue communicating with each other as if the network was singly-connected, it will set up messages circulating indefinitely around the loops and the process most probably will not converge to a coherent equilibrium.

A straightforward way of handling the network of Figure 1 would be to appoint a local interpreter for the loop x_1, x_2, x_3, x_5 that will account for the interactions between x_2 and x_3. This amounts basically to collapsing nodes x_2 and x_3 into a single node, representing the compound variable (x_1, x_2). This method works well on small loops, but as soon as the number of variables exceeds 3 or 4, collapsing requires handling huge matrices and washes away the natural conceptual structure embedded in the original network.

A second method of propagation is based on "stochastic relaxation" [Hinton, Sejnowski and Ackley, 1984]. Each processor interrogates the states of the variables within its influencing neighborhood, computes a belief distribution for the values of its host variable, then randomly selects one of these values with probability given by the computed distribution. The value chosen will subsequently be interrogated by the neighbors upon computing their beliefs, and so on. This scheme is guaranteed convergence, but usually requires very long relaxation times to reach a steady state.

A third method called ***conditioning*** is based on the ability to change the connectivity of a network and render it singly connected by instantiating a selected group of variables. In Figure 1, for example, instantiating x_1 to some value would block the pathway x_2, x_1, x_3 and would render the rest of the network singly connected, where the propagation techniques of the preceding section are applicable. Thus, if we wish to propagate the impact of an observed data, say at x_6, to the entire network, we first assume $x_1 = 0$, propagate the impact of x_6 to the variables $x_2, \ldots, x_5$, repeat the propagation under the assumption $x_1 = 1$ and, finally, linearly combine the two results weighed by the prior probability $P(x_1)$. It can also be executed in parallel by letting each node receive, compute, and transmit several sets of parameters, one for each value of the conditioning variable. This mode of propagation is not foreign to human reasoning. The terms "hypothetical" or "assumption-based" reasoning, "reasoning by cases," and "envisioning" all refer to the same basic mechanism of selecting a key variable, binding it to some of its values, deriving the consequences of each binding separately, and integrating those consequences together.

Finally, an approach is described in Pearl [1984] which introduces auxiliary variables and permanently turns the network into a tree. To understand the basis of this method, consider an arbitrary tree-structured network. The leaves in this network are tightly coupled in the sense that no two of them can be separated by the others, and therefore, if we were to construct a Bayes network with these variables ***alone***, a complete graph would ensue. Yet, together with the intermediate variables, the interactions among the leaf variables are tree structured, thus demonstrating that some networks can be broken up into trees by introducing dummy variables. This scheme enjoys the advantage of uniformity: the processors representing the dummy variables can be identical to those representing the real variables, in full compliance with our architectural objectives. Moreover, there are strong reasons to believe that the process of reorganizing data structures by adding fictitious variables mimics an important component of conceptual development in human beings, the evolution of causal models.

People often invent hypothetical unobservable entities such as "ego", "elementary parti-

cles", and "supreme beings" to make theories fit the mold of causal schema. When we try to explain the actions of another person, for example, we invariably invoke abstract notions of mental states, social attitudes, beliefs, goals, plans, and intentions. Medical knowledge, likewise, is organized into causal hierarchies of invading organisms, physical disorders, complications, clinical states, and only finally, the visible symptoms. Computationally speaking, we can interpret these mental constructs as names given to memory locations that encode a summary of the interaction between the visible variables and, once calculated, permit us to treat the visible variables as if they were mutually independent. Thus, the restructuring of Bayes networks into trees by introducing auxiliary variables shares many computational features with the development of causal models in people. It is suggestive, therefore, to identify the auxiliary variables with the mental constructs of "hidden causes", and to conjecture that humans' relentless search for causal models is motivated by their desire to achieve computational features similar to those offered by tree-structured Bayesian networks.

REFERENCES

Anderson, John R., (1983), "The Architecture of Cognition", Harvard University Press, Cambridge, MA.

Hinton, G.E., Sejnowski, T.J., and Ackley, D.H., (1984), "Boltzman Machines: Constraint Satisfaction Networks that Learn", Technical Report CMU-CS-84-119, Department of Computer Science, Carnegie-Mellon University.

Kim, J. and Pearl, J., (1983), "A Computational Model for Combined Causal and Diagnostic Reasoning in Inference Systems", ***Proceedings of IJCAI-83***, 190-193.

Pearl, J., (1982), "Reverend Bayes on Inference Engines: A Distributed Hierarchical Approach", ***Proc. AAAI Nat'l. Conf. on AI,*** Pittsburgh, PA, pp. 133-136, August.

Pearl, J., (1984), "Learning Hidden Causes from Empirical Data", Technical Report R-38, UCLA Computer Science Dept. To be published in the ***Proceedings of IJCAI-85***.

Shastri, L. and Feldman, J.A., (1984), "Semantic Networks and Neural Nets", TR-131, Computer Science Dept., The University of Rochester, Rochester, NY, June.

Lowrance, J. D., (1982), "Dependency-Graph Models of Evidential Support", COINS Technical Report 82-26, University of Massachusetts at Amherst.

McAllester, D., (1980), "An Outlook on Truth Maintenance", Artificial Intelligence Laboratory, AIM-551, Cambridge: MIT.

Rosenfeld, A., Hummel, A., and Zucker, S., (1976), "Scene Labeling by Relaxation Operations", ***IEEE Trans. on Computers,*** pp. 562-569.

Rumelhart, D. E., (1976), "Toward an Interactive Model of Reading", ***Center for Human Info. Proc. CHIP-56,*** UC San Diego, La Jolla, CA.

Waltz, D. G., (1972), "Generating Semantic Descriptions from Drawings of Scenes with Shadows", AI TR-271, AI Laboratory, Massachusetts Institute of Technology, Cambridge, MA.