DISTRIBUTED BELIEF MAINTENANCE FOR REASONING ABOUT LIKELIHOOD

Judea Pearl

September 1985
CSD-850030

Technical Report
CSD-8500--
R-50
September 1985

Distributed Belief Maintenance for Reasoning About Likelihood

Judea Pearl
Cognitive Systems Laboratory
Computer Science Department
University of California
Los Angeles, CA 90024
judea@UCLA-LOCUS
judea@LOCUS.UCLA.EDU

Submitted to the
Conference on Theoretical Aspects
of Reasoning About Knowledge

Distributed Belief Maintenance for Reasoning About Likelihood

Judea Pearl
Computer Science Department
University of California
Los Angeles, CA 90024

ABSTRACT

Ever since McCarthy and Hayes proclaimed probabilities "epistemologically inadequate" for reasoning with partial beliefs, research in this area has consisted primarily of nonnumerical approaches, attempting to enrich first-order logic with modal operators that capture the notions of default, likelihood, and knowledge. This paper addresses the problem from the opposite extreme: devising new representations to probabilistic models that emphasize the qualitative aspects of the reasoning process and minimize its sensitivity to numerical inputs. We find that, although numbers per se are bad summarizers of implicit knowledge, they can be very useful in processing that which has been explicated. Probabilistic networks of concetpually related propositions, where the numbers serve to regulate and propel the flow of information, allow reasoning about uncertainty to be as knowledge-intensive, accurate, and psychologically plausible as the level of details we care to explicate.

The paper describes a mechanism for maintaining and propagating beliefs in such networks, which facilitates concurrent, distributed, and coherent inferences, and fully conforms to the axioms of probability theory. Using this mechanism as a model of reasoning we find that many arguments against the use of probabilities are no longer valid, while others expose a core of problems that must eventually be confronted by every formalism of partial beliefs.

1. INTRODUCTION

Probability theory is shunned by most researchers in Artificial Intelligence. New calculi, claimed to better represent human reasoning under uncertainty, are being invented and reinvented at an ever-increasing rate. A major reason for the emergence of this phenomenon has been the objective of making reasoning systems *transparent*, i.e., capable of producing *psychologically meaningful* explanations for every intermediate step used in deriving the conclusion. Admittedly, traditional probability theory has erected cultural barriers against meeting this requirement. For example, scholarly textbooks on probability theory create an impression that to construct an adequate representation of probabilistic knowledge we must literally start by defining a *joint distribution function* $P(x_1, \ldots, x_n)$ on all propositions and their combinations, and that this function should serve as the sole basis for all inferred judgments. As a result, even simple tasks such as computing the impact of an evidence e on a hypothesis h via

$$P(h \mid e) = \frac{P(h, e)}{P(e)} = \frac{\sum_{x_i \neq h, e} P(x_1, \ldots, x_n)}{\sum_{x_i \neq e} P(x_1, \ldots, x_n)}$$

appear to require a horrendous number of meaningless arithmetic operations, unsupported by familiar mental processes. Another example is the striking disparity between traditional numerical definitions of independence (e.g. $P(h, e) = P(h) \cdot P(e)$) and the ease and conviction with which people identify conditional independencies, being so unwilling to provide precise numerical estimates of probabilities.

However, other representations of uncertain knowledge are available, which provide a more faithful model of human reasoning, and still comply with the basic tenets of probability theory. *Dependency-graph* representations, in which the links signify direct probabilistic dependencies among semantically-related propositions, are the most appealing candidates because they are robust to numerical imprecisions. They permit people to express essential qualitative relationships and preserve them despite sloppy assignment of numerical estimates. An integral part of dependency-graph models of reasoning is the assumption that the basic steps invoked while people query and update their knowledge correspond to local mental tracings of links in these graphs and this, in turn, determines what kind of operations people consider "psychologically meaningful". *Bayesian networks* offer an effective formalism for describing and controlling such graph operations.

Section 2 summarizes the properties of Bayesian networks and of a Belief Maintenance System (BMS) that performs inferences within such networks [Pearl, 1985a]. The impact of each new evidence is viewed as a perturbation that propagates through the network via local communication among neighboring concepts. We show that in reasonably sparse networks such autonomous propagation mechanism can support both predictive and diagnostic inferences, that it is guaranteed to converge in time proportional to the network's diameter, and that every proposition is eventually accorded a measure of belief consistent with the axioms of probability theory.

Section 3 shows that the current trend of abandoning probability theory as the standard formalism for managing uncertainty is grossly premature--taking graph propagation as the basis for probabilistic reasoning nullifies most objections against the use of probabilities in reasoning systems. For example, the graph representation allows us to:

1. Construct consistent probabilistic knowledge-bases without collecting "massive amounts of data".

2. Admit judgmental evidence at any level of abstraction.

3. Ensure that evidence in favor of a hypothesis not be construed as partially supporting its negation.

4. Postpone judgement.

5. Distinguish between various types of uncertainty.

6. Trace back the sources of beliefs and produce sound explanations.

7. Optimize the acquisition of data.

2. BELIEF MAINTENANCE USING PROBABILITIES

2.1 Bayesian Networks

Bayesian 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).

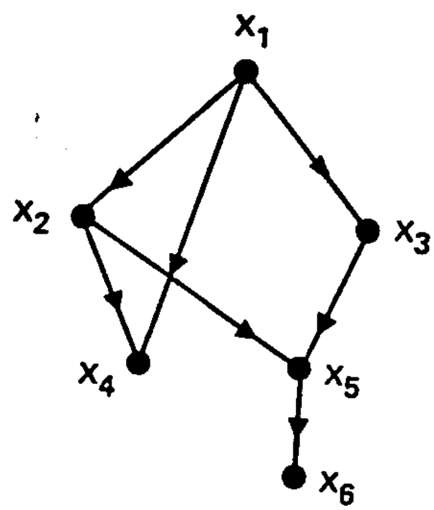

Figure 1

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,

$$P(x_1, \ldots, x_n) = \prod_i P'(x_i \mid S_i)$$

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 the essential independence relationships embedded in the underlying 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. A path is "blocked" if it contains an instantiated variable between two diverging or two cascaded arrows. A different criterion holds for converging arrows: the connection between two arrows converging at node x_k is normally "blocked", unless x_k or any of its descendants is instantiated. 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.

2.2 Belief Propagation in Bayesian Networks

Once a 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 E corresponding to the available evidence and calculating its impact on the probabilities of a set of variables H designated as hypotheses. In principle, this process can be executed by an external interpreter who may have access to all parts of the network and may use its own computational facilities to store and manipulate intermediate results. An extreme example would be to calculate $P(H \mid E)$

using the ratio definition $P(H, E)/P(E)$ (see Introduction). However, the sequence of steps followed by such an interpreter would seem foreign to human reasoning, and would not be defensible by a psychologically meaningful explanation.

A more transparent interpretation process results when we restrict the computation to take place at the *knowledge level itself*, not external to it. That means that the links in the network are the only pathways and activation centers that direct and propel the flow of data in the process of querying and updating beliefs. Accordingly, we imagine 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 examine the messages received from 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 transmit new messages to its neighbors. This, of course, will activate similar revisions at the neighboring nodes and will set up a multidirectional propagation process, until equilibrium is reached.

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 asynchronously?

The key to maintaining stability in bi-directional inference systems lies in storing with each proposition an explicit record of the sources of its belief. Thus, in addition to its measure of total belief, each proposition also maintains a list of parameters, called *support list*, each representing the degree of support that the host proposition obtains from one of its neighbors.

2.3 Maintaining The Support List

The problems associated with asynchronous propagation of beliefs, have simple solutions 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 [Pearl, 1982], as well as graphs with multi-parent nodes, representing events with several causal factors [Kim and Pearl, 1983]. We shall first describe the propagation scheme in singly connected networks and then show how it can be modified to handle loops.

Consider a fragment of a singly connected Bayesian network, as depicted in Figure 2. Let variable names be denoted by capital letters, e.g., A, B, X, Y, and their associated values by subscripted lower case letters, e.g., $a_1, a_2 \cdots$. With each variable A we store the following three parameter lists:

1. $P(A \mid B, C)$ -- The fixed conditional probability matrix which relates the variable A to its immediate causes (parents).

2. $\pi_A(B)$ -- The current strength of *causal* (or *prospective* support, contributed by each incoming link, e.g., from B to A, where:

$$\pi_A(B) = P(B = b_j \mid \text{evidence connected to } A \text{ via } B) \qquad j=1,2,... \tag{1}$$

3. $\lambda_X(A)$ -- The current strength of *diagnostic* (or *retrospective*) support contributed by each outgoing link, e.g., from A to X, where:

$$\lambda_X(A) = P(\text{evidence connected to } A \text{ via } X \mid A = a_i) \qquad i=1,2,... \tag{2}$$

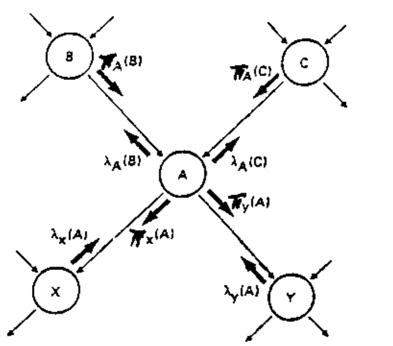

Figure 2

The absolute magnitudes of the elements in each of the λ vectors are arbitrary; only their ratios count. In the case of bivalued (propositional) variables, only a single parameter, the likelihood ratio, is needed. The π–λ parameters are delivered to each node by the corresponding neighbors and are sufficient for calculating the current belief over the values of A:

$$\text{Bel}(A) = P(A = a_i \mid \text{all evidence})$$
$$= f_b\,[P(A \mid B, C), \lambda_X(A), \lambda_Y(A), \pi_A(B), \pi_A(C)]. \tag{3}$$

Similarly, the π–λ parameters stored at A are sufficient for calculating the appropriate messages (also π–λ parameters) that A should deliver to its corresponding neighbors. These calculations involve all the stored parameters except the one obtained from the port receiving the message. For example:

$$\lambda_A(B) = f_\lambda[P(A \mid B, C), \lambda_X(A), \lambda_Y(A), \pi_A(C)], \tag{4}$$

$$\pi_X(A) = f_\pi[P(A \mid B, C), \lambda_Y(A), \pi_A(B), \pi_A(C)]. \tag{5}$$

The combining functions f_b, f_λ, and f_π involve only inner products and component-by-component products [Kim, 1983].

The impact of new evidence propagates through the network by uniform local computations which may be concurrent, asynchronous, or activated by some goal-oriented strategy. Upon receiving an activation signal, each processor examines the π–λ parameters stored, then recomputes and transmits the π–λ messages for its neighbors. Eqs. (4) and (5) demonstrate that a perturbation of the causal parameter, π, will not affect the diagnostic parameter, λ, on the same link and vice versa. The two are orthogonal to each other since they depend on two disjoint sets of data. Therefore, any perturbation of beliefs due to new evidence propagates through the network and is absorbed at the boundary without reflection. A new equilibrium state is reached after a finite number of updates which, in the worst case, is equal to the diameter of the network.

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, and associative retrieval, where unsupervised parallelism is an uncontested mechanism.

Special provisions are necessary to support propagation in networks containing loops (like the one in Figure 1), where parents of common children also possess common ancestors. 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 will not converge to a coherent equilibrium.

The method that we found most promising, called *conditioning* [Pearl, 1985b], 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, 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 paragraphs 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 posterior probability $P(x_1 | x_6)$. 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 these consequences together.

REFERENCES

[1] J. Kim, *CONVINCE: A CONVersational INference Consolidation Engine*. Ph.D. dissertation, University of California, Los Angeles, 1983.

[2] J. Kim and J. Pearl, "A computational model for combined causal and diagnostic reasoning in inference systems", *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, pp. 190-193, 1983.

[3] J.D. Lowrance, "Dependency-graph models of evidential support", COINS Technical Report 82-26, University of Massachusetts at Amherst, 1982.

[4] J. Pearl, "Reverend Bayes on inference engines: A distributed hierarchical approach", *Proceedings of AAAI Conference on Artificial Intelligence* (Pittsburgh, PA), 1982.

[5] J. Pearl, "Fusion, propagation, and structuring in Bayesian networks", Report #CSD-850022, Cognitive Systems Laboratory, Computer Science Department, University of California, 1985a.

[6] J. Pearl, "A constraint-propagation approach to probabilistic reasoning", *Proceedings of the Workshop on Uncertainty and Probability in Artificial Intelligence* (UCLA, 14-16 August 1985), pp. 185-192, 1985b.