Learning Extensible Multi-Entity Directed Graphical Models

Kathryn Blackmond Laskey
Dept. of Systems Engineering and Operations Research
George Mason University
Fairfax, VA 22032-4444
klaskey@gmu.edu

Abstract

Graphical models have become a standard tool for representing complex probability models in statistics and artificial intelligence. In problems arising in artificial intelligence, it is useful to use the belief network formalism to represent uncertain relationships among variables in the domain, but it may not be possible to use a single, fixed belief network to encompass all problem instances. This is because the number of entities to be reasoned about and their relationships to each other varies from problem instance to problem instance. This paper describes a framework for representing probabilistic knowledge as fragments of belief networks and an approach to learning both structure and parameters from observations.

1 INTRODUCTION

A graphical model is a parsimonious and modular parameterization for a joint probability distribution on a set of random variables. We consider models in which the joint distribution is represented as an acyclic directed graph together with a set of local distributions. The nodes in the graph represent random variables and the edges represent conditional independence assumptions satisfied by the joint probability distribution. The local distribution for each node specifies the conditional distribution for its random variable given the values of its parent random variables. The joint distribution on all the variables is then defined as the product of these local distributions. A consequence of this definition is that a random variable is conditionally independent of its non-descendents given its parents.

Belief network models are typically assumed to apply to a population of exchangeable individuals, where all individuals are characterized by the same attributes, the same state spaces for the attributes, the same independence constraints, and the same local functions for their probability models. For example, in a medical diagnosis problem, the population might be a set of patients visiting a clinic and the attributes might be variables relating to medical history, symptoms, test results, and conditions from which the patient might be suffering. In an image restoration problem, the population might be digitized images and the variables might be pixel colors and intensities, edges, pixel texture classifications, etc. This kind of graphical model has been called a *template model*, because the same template (variables, states, arcs, local functions) applies to all individuals in the population. Inferring both structure and parameters of a template model from a sample of individuals is a well-understood problem, although quite challenging and interesting for some problems, such as those with missing data or unobserved variables (e.g., Cooper and Herskovits, 1992; Cooper, 1995; Madigan and Raftery, 1994; Friedman, 1998). Inferring the values of some variables for a given individual conditional on the values of other variables is also a well-understood problem for which general purpose exact and approximate algorithms exist and continue to be refined (Jensen, 1996).

Template models are quite useful for the problems to which they apply and have fostered a surge of interdisciplinary interest in decision theoretic methods. However, many of the most challenging and interesting problems in artificial intelligence cannot be treated with template models. In such problems, the number of individuals to be reasoned about and their relationships to each other vary from problem instance to problem instance. While it may be useful to use the language of graphical models to represent uncertain relationships among variables in the domain, no fixed model with finitely many variables can represent all problem instances. Examples of this kind of domain include natural language understanding (Charniak and Goldman, 1993) and military situation assessment (Laskey and Mahoney, 1997; Mahoney and Laskey, 1998; Laskey, 1997). Although they consider only notional applications, the representation frameworks of Koller and Pfeffer (1997) and Haddawy (1994) are intended to address this kind of problem.

This paper describes a class of models for reasoning about variable numbers of entities related to each other in varying ways, and describes a probability model that can form the basis for a learning algorithm. Section 2 describes the representation framework. Section 3 describes the problem of learning parameters for a fixed model structure, and Section 4 addresses the problem of learning the structure. Section 5 discusses open research issues.

2 EXTENSIBLE MULTI-ENTITY MODELS

It is useful to be able to specify and store a model in terms of modular components which are assembled at run-time to create a Bayesian network to reason about a particular problem instance. Laskey and Mahoney (1998) define a class of models called *extensible multi-entity (EME) models*. An entity is an individual or object about which the model reasons. In a standard template model, the sample space is the set of entities (e.g., patients) and the random variables map entities to the values of their attributes (e.g., symptoms of patients). Template models can also be defined over a sample space of tuples of entities, when the number of entities and their relationships to each other are fixed. For example, in the military situation awareness domain, a model for reasoning about a missile unit might contain random variables for attributes of the unit (e.g., the distance it can shoot), but might also contain random variables describing attributes of related units (e.g., the type of unit it is defending) or relationships between the unit and other units (e.g., its distance from the unit it is defending). An extensible multi-entity model is designed for problems in which the number of entities being reasoned about for any problem instance, and the roles entities play in relation to other entities, are not fixed in advance. For example, the reasoner might need to consider a situation in which there is an unknown number of units, the types of the units may be unknown, and it may not be known which units are defending which other units. The template missile unit model described above is conditioned on the type of the unit and the identity (although not necessarily the type) of the unit it is defending. This model could form part of a knowledge base of model fragments that are pieced together at run-time to form a global model for a complex situation in which it is unknown how many missile units there are, or which unit any given missile unit is defending.

Extensible multi-entity models are designed for this type of problem. Knowledge about entities and their relationships is represented as a set of directed graphical model fragments, which are assembled at run-time to construct a model to reason about a given problem instance. This paper summarizes the definitions and model construction approach described in Laskey and Mahoney (1998), and then considers the problem of inferring the structure and parameters of extensible multi-entity models from a sample of observations.

Entities are categorized into types (e.g., people, trucks, missile units). All entities of a given type are assumed to be exchangeable. A model is represented as a collection of partially specified directed graphical models called *network fragments*. The nodes of a network fragment represent random variables referring either to attributes of entities or to relationships among tuples of entities. The fragment defines distributions for some of its variables, called *resident* variables. Variables that condition the distributions of resident variables, but which are not themselves resident in a fragment, are called *input* variables. The following definitions are from Laskey and Mahoney (1998).

Definition 1: Let T be a set consisting of a finite number of entity types. For each *t*   *T*, let $\surd_t$ be a set of possible values for entities of that type. An *entity* is a pair *e*=(*t*,*v*), where *t*   T and *v* $\surd_t$.

Definition 2: A *random variable* is a proposition or question about a tuple of entities that can take on values in a mutually exclusive and collectively exhaustive set of outcomes in a finite *outcome set.* Each random variable is uniquely identified by a name, a tuple of finitely many *roles* with associated types, an entity of appropriate type assigned to each role, and an outcome set.

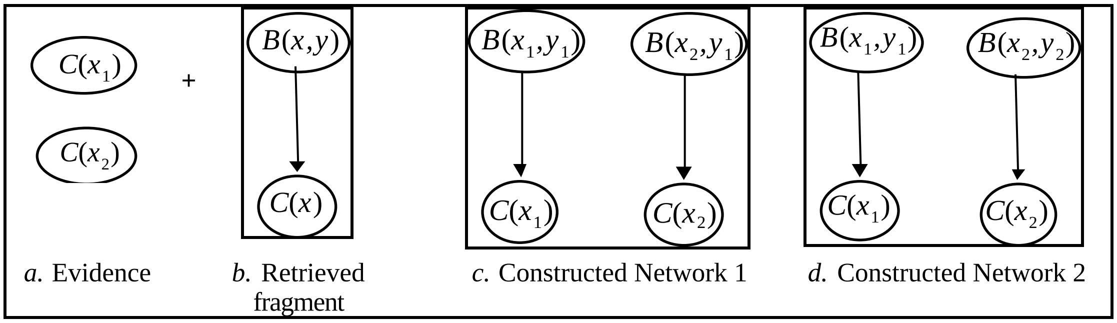
Classes of exchangeable random variables can be defined by allowing the entities filling a fragment's roles to vary over entities of the appropriate types.

Definition 3: A *random variable class* is identified by a name, a set of roles, and a type for each role. The class consists of all random variables obtained by assigning entities of appropriate types to the roles.

It is useful to permit entity types to be arranged in a directed graph called a type hierarchy. Random variable roles may then be filled by subtypes of their associated entity type. If types are arranged in a type hierarchy, only the primitive types (those with no subtypes) are assumed to be exchangeable. For supertypes, the distribution of the variable may depend on the entity subtype. This is represented by conditioning the distribution of the random variable on a random variable representing the subtype.

To define an EME model, we need to specify a consistent assignment of probability distributions to random variables in a way that satisfies the exchangeability assumption for type-consistent assignments of entities to roles in random variable classes. This is accomplished by defining classes of network fragments. Fragments assign conditional distributions to their resident random variables. Like random variable classes, fragment classes have roles to which type-consistent assignments of entities can be made.

Definition 4: A *network fragment class* represents knowledge about the conditional distribution of a set of random variable classes. A network fragment class



a. Evidence *b.* Retrieved fragment *c.* Constructed Network 1 *d.* Constructed Network 2

Figure 1: Multiple Hypotheses for Network Construction

consists of a name, a finite set of roles, a type for each role, a set of input random variable classes, a set of resident random variable classes, a set of role correspondence functions, a fragment graph, and a set of local distributions. A role correspondence function is associated with each random variable in the fragment, and specifies a fragment role for each of the random variable roles. The mapping of random variable roles to fragment roles is assumed to be one-to-one and type consistent. The fragment graph is an acyclic directed graph with a node for each input and each resident random variable class. Nodes corresponding to input variables must be roots in the graph. With each resident variable is associated a set of probability distributions on its state space. For resident random variables that are roots in the graph, there is one such distribution, called the marginal distribution. For non-roots, there is a distribution for each combination of states for its parent random variable classes.

Every type-consistent assignment of entities to roles in the network fragment specifies an *instance* of the fragment class. A fragment instance assigns entities to fragment roles, inducing an assignment to random variable roles via the role correspondence function. All instances of a fragment encode identical conditional distributions on their resident random variables given their input random variables. The following conditions given in Laskey and Mahoney (1998) ensure that a set of network fragments is consistent with at least one template model.

Proposition: Consider a finite set of random variable classes and a finite set of network fragment classes defined on these random variable classes. Suppose the following conditions are satisfied:

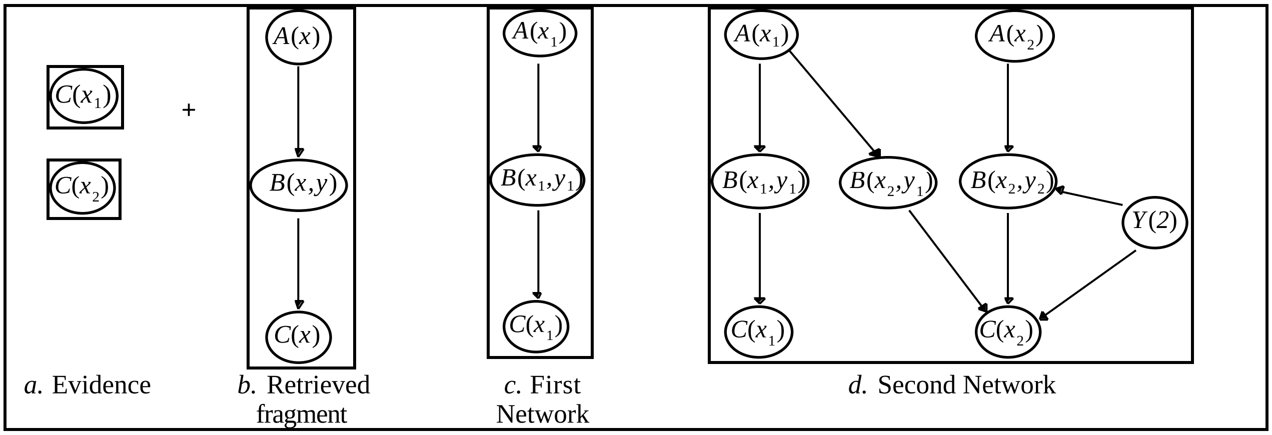
1. Each random variable class, identified by its name, roles, and role types, is resident in no more than one network fragment.
2. Each random variable class appearing in some fragment is resident in at least one fragment.
3. The graph union of all the network fragments contains no directed cycles. In performing the graph union, the identity relation is the random variable class.

Then any function that maps entities to network fragment roles in a type-consistent manner defines a template Bayesian network model, and at least one such assignment exists.

Condition 3, acyclicity, would be violated in temporal models, where it is common for random variables at a given time to condition the same random variables at a later time. An extension of this framework is required for temporal models.

To develop the theory of EME models, we need to introduce additional machinery to support inference about which entities are being reasoned about and how they fill the fragment roles. Specifically, consider the situation in which a fragment role is not mentioned in a random variable but is mentioned in one of its parents, as for $C(x)$ in Figure 1*b*. When an instance $C(x_1)$ is created, the conditioning events that should be used to define its distribution are ambiguous until the entity filling the y role in $B(x,y)$ is specified. If there are no already hypothesized entities to fill the y role, a new entity y_1 is created. Now, when a second instance $C(x_2)$ is created, there are two possibilities for the y role in its parent: the already hypothesized entity y_1 and a previously unobserved entity y_2. These hypotheses are illustrated in Figures 1*c* and 1. Reasoning with extensible multi-entity models requires specifying a probability model for when to create entities and how to assign entities to roles.

A *situation*, or problem instance, is defined as a set of random variable instances called the *mentioned* random variables, together with values for a subset of the mentioned variables, called the *evidence* variables. For example, the situation referred to in Figure 1 consists of mentioned variables $C(x_1)$ and $C(x_2)$, together with their values if they are evidence variables. The entities filling the roles in the mentioned variables are all assumed to be specified explicitly (i.e., the particular entities x_1 and x_2 are specified as part of the situation). The objective of network construction is to compute a probability distribution for the mentioned variables, conditional on the reasoner's information about the situation. This information includes the values of the evidence variables, but it also includes which variables are mentioned. Although most research in belief network learning and inference ignores the process by which situations are generated, both network construction and



a. Evidence *b.* Retrieved fragment *c.* First Network *d.* Second Network

Figure 2: Extensible Multi-Entity Model (second evidence)

learning for extensible multi-entity models require an explicit model of how variables become mentioned. The following assumptions underly the network construction approach summarized here:

1. Conditional on the number of entities of each type and the assignment of entities to roles, the joint distribution on random variable instances is independent of which variables are mentioned, and is defined by the local distributions specified in the network fragments.
2. Conditional on the number of entities of each type, all type-consistent assignments of entities to undesignated roles are *a priori*[1] equally likely.
3. Entities appear in the situation (i.e., appear either in a mentioned random variable or an ancestor of a mentioned random variable.)[2] independently of each other with a type-specific entity existence probability τ_t. That is, the number of entities of a given type in a situation follows a binomial distribution with parameter τ_t.

When the model is created by eliciting structure and parameters from domain experts, it is incumbent on the modeler to define the model so that conditions 1 through 3 are met. Sometimes this requires explicit modeling of observability of entities, and inclusion of random variables that affect whether other random variables will be observed.

Laskey and Mahoney (1998) describe an algorithm to construct a situation-specific belief network to infer distributions for non-evidence mentioned variables given evidence variables in extensible multi-entity models. A situation-specific network (Mahoney and Laskey, 1998) is a belief network that contains only those variables needed to compute the distribution of target variables given evidence variables.

The construction algorithm is illustrated in Figure2. The evidence $C(x_1)$ causes the fragment in Figure 2*b* to be retrieved. The entity y referred to by random variable $B(x_1)$ remains unspecified. Because no entities of type y have yet been hypothesized, we hypothesize a new entity y_1, create random variable instance $B(x_1,y_1)$ and obtain the model of Figure 2*c*. Next, evidence variable $C(x_2)$ is introduced. There are now two possible referents for y: the entity y_1 already hypothesized, and an entity y_2 hypothesized to explain the new evidence. An *existential random variable* $Y(2)$ is created with values y_1 and y_2 to represent these two possibilities. Variable $C(x_2)$ takes its distribution conditional on the values of $B(x_2,y_1)$ if $Y(2)=y_1$ and conditional on the values of $B(x_2,y_2)$ if $Y(2)=y_2$. $Y(2)$ is a parent of $B(x_2,y_2)$ because the latter variable only exists under the hypothesis that y_2 exists as a separate entity distinct from y_1. Both $B(x_2,y_2)$ and $C(x_2)$ take on the special value * (meaning nonexistent) if the value of $Y(2)$ is y_1. The final model is shown as Figure 2*d*. Note that the algorithm defined in Laskey and Mahoney (1998) yields different, graph structures when variables are processed in a different order, but the different structures all yield the same joint distribution on the mentioned variables.

3 LEARNING PARAMETERS GIVEN STRUCTURE

Consider the problem of inferring an EME model for a set of random variable classes defined on a set of entity types. The structure of the model includes: (1) the fragment graphs; (2) constraints and functional forms for the local distributions in the fragment graphs; and (3) the role correspondence functions. The parameters include parameters of the local distributions in the fragments, together with the type-specific entity existence probabilities.

The first step in defining a learning approach is to consider the problem of inferring parameters of an extensible multi-entity model when the structure is known. Given a set of situations sampled as described in Section 2 from an extensible multi-entity model of given structure, one first computes a situation-specific

[1] Prior to conditioning on the observed values of the evidence variables.

[2] Note that the joint distribution of the mentioned random variables is independent of any variables that are not either mentioned variables or ancestors of mentioned variables.

belief network for each of the situations. If the purpose is learning, it makes sense to assume that the states of all mentioned variables are known. Any unmentioned variables created during network construction are unobserved, as are all the existential variables. Thus, parameter learning for EME models is a problem of inference with incomplete data and hidden variables.

For each random variable class, a sufficient statistic for its parameter distribution is the aggregated set of counts, across all instances of the random variable class in all the sampled situations, of combinations of values of the variable and the parents indicated by the value of its existential variables (if any). Because many of the random variables, including the existential variables, are unobserved, the sufficient statistic is generally not fully observed. A missing data estimation method such as EM or Gibbs sampling can be used to estimate posterior distributions for the parameters (e.g., Cooper, 1995; Friedman, 1998).

The complete-data sufficient statistics for the type-specific existence probabilities are the counts of entities of each type. These sufficient statistics too are not fully observed. Again, the posterior distribution can be estimated using EM or Gibbs sampling.

Posterior distributions for high-dimensional problems with missing data and hidden variables can be extremely difficult to estimate. Typically the parameter space is multimodal. Hill-climbing algorithms such as EM tend to become stuck in local optima. MCMC approaches, although they can escape local optima, can be extremely slow to converge. Algorithms for this type of learning problem are an active area of research.

4 LEARNING STRUCTURE

The structure learning problem can be simplified by assuming that each fragment contains only a single resident random variable, and defines exactly one local distribution. Therefore, there is no issue of learning which variables belong together in a fragment or what the fragment role correspondence functions are. The only structure learning problem is to learn which variables to include as input variables to each fragment, or, equivalently, the global belief network graph. Most belief network learning algorithms assume a uniform prior over graphs, although more complex prior distributions are possible. Structure learning algorithms typically consist of two components: search and scoring. The search algorithm enumerates structures using a heuristic designed to find good structures. Different scoring metrics have been proposed, most of which can be considered as approximations to the log posterior probability of the observations given the structure. The score is multiplied added to the log prior probability of the structure to obtain a value equal, up to an additive constant, to the log posterior probability of the structure. Most algorithms return the highest-scoring structure, although some authors have suggested using a weighted average of several high-scoring structures.

When there are no missing data and a conjugate prior distribution is used, the predictive probability of the sufficient statistics can be obtained in closed form (e.g., Cooper and Herskovits, 1992). For more complex problems the predictive distribution must be approximated. A common approach is to use Laplace's method, which is based on a normal approximation to the log-likelihood function centered at the posterior mode of the parameter distribution. The accuracy of this approximation improves as the number of observations for each parameter becomes larger. Another approach to estimating the predictive probability of the observed data is to use a Markov chain Monte Carlo method.

There is no new theory required for extending structure learning to the problem of inferring which clusters of random variables should be grouped together to form fragments. All that is required is a prior distribution for variable groupings and role correspondence functions. Grouping variables together in a fragment induces constraints on the role correspondence. For example, in Figure 1, the same entity must fill the x role in random variables $A(x)$, $B(x,y)$ and $C(x)$. Often there is strong *a priori* knowledge about which random variables should be grouped together in a fragment. Such knowledge can be used to improve the efficiency of a learning algorithm.

5 DISCUSSION

A class of models called extensible multi-entity (EME) models, was introduced for problems in which the number of entites and their relationships to each other may be uncertain. EME and other similar kinds of probability model show promise to extend the reach of decision theoretic methods to problems of far greater complexity than could previously be treated. The representation, the underlying sampling model, and an algorithm for constructing situation-specific belief networks were briefly reviewed.

If these models are to find wide application, it is necessary to move beyond the tedious and time-consuming knowledge engineering bottleneck. The methods described in this paper are a step in this direction. Computing posterior distributions for parameters and structure given a data set of situations involves relatively minor modifications of existing algorithms for learning belief networks in problems with missing data and hidden variables.

This type of problem is among the most difficult of learning problems because the parameter space tends to be multimodal. Further research is needed to develop efficient parameter estimation algorithms for complex multimodal distributions and efficient search methods for structures.

Acknowledgments

The research reported in this paper was sponsored by DARPA and the Air Force Research Laboratory under contract DACA76-93-0025 to Information Extraction and Transport, Inc., with subcontract to George Mason University. Many thanks to Suzanne Mahoney and Bruce d'Ambrosio for helpful and stimulating discussions.

References

Charniak, E. and Goldman, R (1993) A Bayesian Model of Plan Recognition, *Artificial Intelligence 64*, p. 53-79.

Cooper, G.F. (1995) A Bayesian Method for Learning Belief Networks that Contain Hidden Variables. *Journal of Intelligent Information System 4*, p. 71-88.

Cooper, G.F. and Herskovits,E. (1992). A Bayesian Method for the Induction of Probabilistic Networks from Data. *Machine Learning 9*, p. 309-347.

Friedman, N. (1998) The Bayesian Structural EM Algorithm. *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Morgan-Kaufmann.

Haddawy, P. (1994) Generating Bayesian networks from Probability Logic Knowledge Bases. In Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers, San Francisco, CA pp. 262-269.

Jenssen, F. (1996) *An Introduction to Bayesian Networks*. New York, Springer.

Koller, D. and A. Pfeffer (1997) Object-Oriented Bayesian Networks In Geiger, D. and Shenoy, P. (eds)*Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference,* San Francisco, CA: Morgan Kaufmann.

Laskey, K.B. (1997) *Recognition Fusion: Network Fragments for Inferring Entity Existence and Type*, Rosslyn, VA: Information Extraction and Transport, Inc.

Laskey, K.B. and S. M. Mahoney (1997) Network Fragments: Representing Knowledge for Constructing Probabilistic Models. In Geiger, D. and Shenoy, P. (eds)*Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference*, San Francisco, CA: Morgan Kaufmann.

Laskey, K.B. and S. M. Mahoney (1998) Constructing Graphical Models by Combining Model Fragments, submitted to *Proceedings of the 6th Annual Valencia Meetings on Bayesian Statistics*.

Madigan, D. and Raftery, A. (1994) Model Selection and Accounting for Model Uncertainty in Graphical Models using Occam's Window. *Journal of the American Statistical Society 89* p. 1335-1336.

Mahoney, S.M. and Laskey, K.B. (1998) Constructing Situation-Specific Networks. In Cooper, G. and Moral, S. (eds) *Uncertainty in Artificial Intelligence: Proceedings of the Fourteenth Conference*, San Francisco, CA: Morgan Kaufmann.