Probabilistic Relation Networks

Erik Skarman

June 26, 2003

Introduction

A relation network is a form of a model of a system in the real world. The model is a graph with nodes and arcs, where a node represents an entity, i.e. it represents what we know about that entity, and the arcs represent relations between the node entities.

Given the relations between the node entities, the knowledge of one node entity, can be mapped over to a corresponding knowledge of another. This process is called *transport*.

Typically, more than one arc leads to one node. Then it is possible to transport knowledge from more than one neighboring node to that node. There is then a process called *unification* in which these different pieces of knowledge are negotiated together into a unified knowledge about the node entity.

In a set-based relation network, the data stored in a node is a set, in which we believe that the value of the entity is. Transport through relations which can be expressed by functions then amounts to mapping sets through these functions. Unification is set intersection.

Here, we will discuss probabilistic relation networks, in which the node entities are considered as stochastic variables, and in which this stochastic nature is represented with probability densities. Transport through relations here amounts to mapping probability densities through relations. Unification is a multiplication-like process which we will come to.

In the final sections we will return to the set representation and a representation using predicate calculus.

Probability Notations

We denote stochastic variables with capital letters, X, Y etc. Small letters like x, y etc denote numbers. These numbers may be values that the stochastic variables can take.

The letter P (capital letter) represents the likelihood of an event written in parentheses: P(event).

An event may be that a continues stochastic variable X, takes a value x. We could write that P(X=x), but that likelihood is usually zero, as it is unlikely that

X would be exactly x. Instead, we would like to express a likelihood density, or likelihood per volume. That is the likelihood that X falls in some set M, divided by the volume Vol(M) of M. We define:

$$p_X(M) = \frac{P(X \in M)}{Vol(M)} \tag{1}$$

If M shrinks to a point x, this will converge to a limit which we call $p_X(x)$. Conversely then, we can compute $P(X \in M)$ as

$$P(X \in M) = \int_M p_X(x) \cdot dV \tag{2}$$

The function p_X is called the probability density of X.

Conditional Likelihoods

If we assess the likelihood of an event E, while we have some knowledge, K, we get to another result, called $P(E|K)$, than if we did not have that knowledge. $P(E|K)$ is called the conditional likelihood of E, given K.

Thomas Baye gave the following definition of conditional likelihood, which is consistent with the meaning of the term, that we just gave:

$$p(E|K) = \frac{p(E,K)}{p(K)} \tag{3}$$

Here we think of the "knowledge" K in the following way: What we know is that an event K has happened. p(E,K) is the likelihood that E and K happen simultaneously (the simultaneous likelihood).

Two events are said to be *independent* if $p(E|K) = p(E)$. From this it follows that, in that case E and K are independent, $p(E,K) = p(E) \cdot p(K)$.

Quite in general there is the symmetry that $p(E,K) = p(K,E)$, from which it follows that:

$$p(E|K) = \frac{p(K|E) \cdot P(E)}{P(K)} \tag{4}$$

This is called Baye's rule, and it is quite important. It allows us to turn the conditioning around. It is the theoretical framework behind the famous Kalman filter, and there are several other schemes for data fusion where it appears. In fact, as soon as we base data fusion on notions of probability and stochastic variables, Baye's rule is almost indispensable. Baye's rule is also helpful when one tries to generalize Kalman filters to cases where Kalman filter theory properly does not apply. A special kind of data fusion scheme is directly named after this rule, and is called Bayesian networks. We will investigate the relationship between Bayesian networks and the relation networks presented here later.

Just as there can be conditional probabilities, there can be conditional probability densities, like $p_X(x|K)$, and there is an equivalent of Baye's rule also for these.

Also quite in general, given the simultaneous likelihood P(E,K) we can get the likelihood for just E, by summing over all options for K. That gives us:

$$P(E) = \sum_K P(E, K) = \sum_K P(E|K) \cdot P(K) \tag{5}$$

Node Representation

As long as a node entity is a continuous variable, the knowledge about it is represented with probability densities $p_X(x)$. For a discrete variable we can go back to the conceptually simpler function P(X=x), for which we may however use the same notation $p_X(x)$.

However, for the unification process, we have a need to keep track of independence, and that requires that we keep track of the information used for computing $p_X(x)$. Hence, all probabilities are assumed to be conditional with the information that has been used for computing them. At some stage of computation, the information K has been used. At that stage, it is $p_X(x|K)$ we know, and that is something we have to know for the future.

Then we need to store the values of the function $p_X(x|K)$ *together with* some information about the knowledge K.

Exactly how this is done is a technical issue, but at least when we develop equations, we should keep this in mind.

Transport of Probabilities through Functions

Assume that, for two stochastic variables X and Y, it is known that Y=f(X). This means that X and Y "are not stochastic for each other", so that given $p_X(x)$, $p_Y(y)$ is also given. For that we have the following equation:

$$p_Y(M) \cdot Vol(M) = p_X(f^{-1}(M)) \cdot Vol(f^{-1}(M) \tag{6}$$

Both members are the likelihood of the same event, namely that Y falls in M, alias that X falls in $f^{-1}(M)$. In the limit when M converges towards {y}, this converges to

$$p_Y(y) = D(f^{-1}) \cdot p_X(f^{-1}(y)) \tag{7}$$

where $D(f^{-1})$ is a volume scaling function. It can be computed as the determinant of the Jacobian matrix of f, which is define by

$$J_{ij} = \frac{\partial f_i^{-1}}{\partial X_j} \tag{8}$$

This is the formula that we need to transport probabilities through relations, which can be determined by functions. The formula applies equally well for conditional densities, where p_X and p_Y are conditioned by the same event. In other words, when we transport knowledge from one node to another, we take the conditions with us.

Relations with uncertainty

In the last section, the variables X and Y determine each other completely, and thus the densities determine each other completely. We can think of all sorts of relations, where the relation between X and Y can involve a stochastic element, so that we don't know exactly what Y is, even though we know exactly what X is.

A prototype for this type of relation is the following:

$$Y = X + E \tag{9}$$

where E is a stochastic variable. This is really a ternary relation (a relation between three) between X, Y and E. We can handle this with the function concept as in the previous section, but we will prefer here to regard E merely as a disturbance, that we are not really interested in determining. The computation of $p_Y(y)$ now goes over the conditional probability $p_Y(y|x)$. If we know that X=x, and assume that Y=y, then the explanation must be that

$$e = y - x \tag{10}$$

and this happens with probability $p_E(y - x)$. So we have

$$p_Y(y|x) = p_E(y - x) \tag{11}$$

Then, we get $p_Y(y)$ from the argument that

$$P(E) = \sum_K P(E, K) = \sum_K P(E|K) \cdot P(K) \tag{12}$$

as

$$p_Y(y) = \int p_E(y - x) p_X(x) dx \tag{13}$$

This can then be generalized for the case, that the relation is

$$Y = f(X, E) \tag{14}$$

There is an inverse of f, for which we borrow the notation f^{-1} which computes e, given y and x. Then we get

$$p_Y(y) = \int p_E(f^{-1}(y, x)) p_X(x) dx \tag{15}$$

Unification

Any likelihood can always be thought of as a conditional likelihood, being conditioned with some knowledge that we didn't think of or took for granted. Conversely, we can always regard a conditional likelihood as an unconditional one, by taking the given knowledge as granted. This leads us to an expansion of Baye's rule in the following manner. First let us note that the denominator in Baye's rule only is a factor to normalize the density, so that the total probability is one. We call this normalizing denominator "norm". Then we have Baye's rule

$$p(E|K) = \frac{p(K|E) \cdot p(E)}{norm} \tag{16}$$

If we now consider everything conditioned with an event L, we have

$$p(E|K, L) = \frac{p(K|E, L) \cdot p(E|L)}{norm} \tag{17}$$

This maps $p(E|L)$ which is what we knew about E as we only knew L, to $p(E|K, L)$ which is what we know about E, now when we know both K and L.

The crucial point is now, that if K is independent of L then

$$p(K|E, L) = p(K|E) \tag{18}$$

Then, we can compute $p(K|E, L)$ without knowing anything about L. In this case we have:

$$p(E|K, L) = \frac{p(K|E) \cdot p(E|L)}{norm} \tag{19}$$

We can rephrase this in terms of densities for a stochastic variable X as

$$p_X(x|K, L) = \frac{p_X(K|x) \cdot p_X(x|L)}{norm} \tag{20}$$

The notation $p_X(K|x)$ is new here, but it is generated as a limit of a

$$p_X(K|M) = \frac{P(K|x \in M)}{Vol(M)} \tag{21}$$

Equation (20) tells us not only how to compute the value of the left member, but also what that left member *is*. In the node for X, we replace $p_X(x, L)$ with $p_X(x, K, L)$ using (20).

This is the unification.

Time Context Switching

There is no universal relation between the value of some variable at a time t and the value of that same variable at t+T. A dynamical system is a system, for which such a relation is known for an array of the variables, called state

variables. We reserve the name x for this array. It is usually assumed that the relation can be expressed with a function, so that we have:

$$x(t+T) = f(x(T)) \tag{22}$$

(More generally the state can be considered to be a point in a *state space*, but if that state space can be coordinatized with a number of coordinates, then these coordinates can be taken as state variables, and they can be structured in an array).

Now we can build a system, with one node for each value $x(t_0+nT)$. But we usually economize this by just having one node for each variable. Then we make a time context switching, in which all the nodes simultaneously are supposed to represent the values at t+T rather than t. For the state variables, this means that the value x is replace with f(x). For all other variables, the values become unknown. In a stochastic representation $p_X(x)$ is replaced by $p_{f(X)}(x)$. How this is computed has been described before.

A way to organize this is assign nodes x^+ for state variables x at the next time context. We compute the contents of the nodes x^+ the usual way using the transport mechanism. At time context switching we move the content directly from x^+ to x . x^+ should not be considered to be a state variable, so it should be set to unknown at time context switching.

The Kalman Filter problem

Here, we have a dynamical system with an array X of state variables, for which it holds that

$$X(t+T) = f(X(t)) \tag{23}$$

There is a measured entity Y, which may also be an array of several variables, and which is related to x with:

$$Y = h(X) \tag{24}$$

Then there are actual measurements Y_m ("m" stands for measured), and these are supposed to be related to Y with

$$Y_m = Y + E \tag{25}$$

where E is a stochastic variable.

What now happens is that we observe values y_m on Y_m We denote by Σ_m the sequence of these observed values y_m up till now, and we divide this sequence into

$$\Sigma_m = (\Sigma_m^-, y_m) \tag{26}$$

In these terms, we want to compute the likelihood $p_X(x|\Sigma_m)$. With $p_X(x|\Sigma_m^-)$ given from the previous step, we can use our unification procedure:

$$p_X(x|\Sigma_m) = \frac{p_X(Y_m = y_m|x) \cdot p_X(x|\Sigma_m^-)}{norm} \tag{27}$$

So what we need now is the likelihood $p_X(Y_m = y_m|x)$. That $Y_m = y_m$ is an event with a nonzero likelihood. But for the conditioning we can't use the event X=x, so we have to work with densities, and that is why X appears as an index in p_X. p_X is thus a density, or a likelihood per volume, and then we get by the same arguments as before that:

$$p_X(Y_m = y_m|x) = D(h) \cdot p_Y(Y_m = y_m|h(x)) \tag{28}$$

Assuming the event that $Y_m = y_m$, and given a value h(x) on Y, we must conclude that E has been

$$E = Y_m - Y = y_m - h(x) \tag{29}$$

That happens with likelihood $p_E(y_m - h(x))$, which is thus the likelihood $p_Y(Y_m = y_m|h(x))$.

$$p_X(Y_m = y_m|x) = D(h) \cdot p_Y(Y_m = y_m|h(x)) \tag{30}$$

For the complete unification computation we get:

$$p_X(x|\Sigma_m) = \frac{D(h) \cdot p_E(y_m - h(x)) \cdot p_X(x|\Sigma_m^-)}{norm} \tag{31}$$

Bayesian nets

Bayesian nets, which are described in the literature, are similar to probabilistic relation networks. They have nodes corresponding to variables, and arcs corresponding to relations. But the relations are describes solely with conditional likelihoods, while in the relation nets, relations are described with mathematical concepts. In most cases, the user recognizes the relations in some mathematical form, and for use in a Bayesian network, he has to compute the conditional likelihoods from his knowledge about the relation. It seems, though, that many designers of Bayesian networks are not aware of this connection with a mathematical relation, but rather "invent" tables of conditional likelihoods manually. On the other hand, there are cases when the conditional likelihood can be chosen more flexibly, when one feels free from tying the likelihoods to a specific relation. This is particularly true when relations describe things like a human reacting on the value of some variable.

How conditional likelihoods are computed, given a mathematical relation, has been described before. As an example, if the relation between X and Y is

$$X = Y + E \tag{32}$$

then

$$p_X(x|y) = p_E(x - y) \tag{33}$$

Bayesian networks are usually described only for variables which take discrete values. But there is no difference here with the probabilistic relation networks. When a variable takes continuous values, all functions (including density functions) have to be approximated with functions over a discrete grid.

Set representation revisited

In the set representation of a relation network, every node contains a set, like X which represents the belief that the variable x is contained in X:

$$x \in X \tag{34}$$

If x is known to be related to y so that x = f(x), then the information that $y \in Y$ can be transported to a "bid" on x through

$$X_Y = f(Y) \tag{35}$$

where f(Y) is the standard mathematical notation

$$f(Y) = \{f(y)|y \in Y\} = \{x|\exists y \in Y : x = f(y)\} \tag{36}$$

In words: We let the variable y scan over Y, and see what values f(y) hits. X_Y is now a bid on x, and we unify it with the a priori information X with:

$$X = X \cap X_Y \tag{37}$$

Here, unlike in the probabilistic representation, we don't need to keep track of independence, and we can repeat the unification as many times as we like. In fact we can do unification all over the network in anarchy or in parallel, and when we have done so long enough, then all sets will have converged to stable values, which represent the best estimate of the variables we can get.

A unification can also result in the empty set, which discards some hypothesis we have made before. This mechanism can e.g. be used to classify objects. We can also do things like computing the graph of a relation, which sometimes can be regarded as a way of learning.

Even though the sets X, Y etc, basically represent belief or uncertainty, we can also use sets to represent extensions of things, like the extension of water in an image, or the extension of lakes in a geographical map. If W and L are such sets we can use the relation

$$X + W \subseteq L \tag{38}$$

where X is the set of displacements of W such that the displaced W is contained in L.

In general, set theory is a large theory, which opens many possibilities. Most of them can be lifted over to the probabilistic representation, but often with some difficulty, and it is easier to come up with good ideas, when working with the set representation.

Predicate representation

The set representation can also be cast in the language of predicate calculus.

A predicate P(x) is function, whose value i true for for some values of x and false for others.

A node for the variable x now contains a predicate P(x), with the interpretation that x is believed to take any value for which P(x) is true. Such a predicate has to be expressed in some mathematical terms. One alternative is:

$$P(x) = x \in X \tag{39}$$

This links the predicate representation to the set representation described in the previous section.

There are also 2-ary and n-ary predicates, like R(x,y) and they represent relations. So R(x,y) is true if and only xRy for the corresponding relation R.

Now let us have two variables, x and y with predicates $P_x(x)$ and $P_y(y)$ which represent a priori information about x and y. Let x and y be related by a relation represented by the 2-ary predicate R(x,y). Given the information about y, we can sharpen our information about x by updating $P_x(x)$ to $P_x^+(x)$ as:

$$P_x^+(x) = \exists y(P_y(y) \wedge R(x,y) \wedge P_x(x)) \tag{40}$$

(For $P_x^+(x)$ to be true, $P_x(x)$ has to be true, and there has to exist a y for which $P_y(y)$ is true, for which the relation between x and y holds.)

This equation represents both transport and unification. Once it has been done for x it can be done for y, but now with $P_x(x)$ replaced with $P_x^+(x)$

Let us express $P_x(x)$ and $P_y(y)$ with sets. If the relation between x and y is x=f(y), then the 2-ary predicate R(x,y) can be expressed easily as

$$R(x,y) = (y = f(x)) \tag{41}$$

That gives

$$P_x^+(x) = \exists y(x \in X \wedge y = f(x) \wedge y \in Y) \tag{42}$$

Now it is clear which value of y it is that exists, and makes the expression true. It is of course y=f(x). So we can substitute y with f(x). That takes the predicate y=f(x) into f(x)=f(x) which is a tautology, so it disappears:

$$P_x^+(x) = x \in X \wedge f(x) \in Y \tag{43}$$

f(x) is in Y if x is in $f^{-1}(Y)$ so we get

$$P_x^+(x) = x \in X \wedge x \in f^{-1}(Y) = x \in X \cap f^{-1}(Y) \tag{44}$$

quite in agreement with what we would get from the set representation.