On Integrating Ontologies with Relational Probabilistic Models

Chia-Li Kuo and **David Poole**
Department of Computer Science
University of British Columbia
Vancouver, BC, Canada V6T 1Z4
`{kuo000, poole}@cs.ubc.ca`

Abstract

We consider the problem of building relational probabilistic models with an underlying ontology that defines the classes and properties used in the model. Properties in the ontology form random variables when applied to individuals. When an individual is not in the domain of a property, the corresponding random variable is undefined. If we are uncertain about the types of individuals, we may be uncertain about whether random variables are defined. We discuss how to extend a recent result on reasoning with potentially undefined random variables to the relational case. Object properties may have classes of individuals as their ranges, giving rise to random variables whose ranges vary with populations. We identify and discuss some of the issues that arise when constructing relational probabilistic models using the vocabulary and constraints from an ontology, and we outline possible solutions to certain problems.

Introduction

In many domains, we want ontologies to define the vocabulary and probabilistic models to make predictions. For example, in geology, the need to define standardized vocabularies becomes clear when one looks at detailed geological maps, which do not match at political boundaries because jurisdictions use incompatible vocabularies. There has been much recent effort to define ontologies for geology and use them to describe data (e.g., `http://www.onegeology.org/`). Geologists need to make decisions under uncertainty, and so need probabilistic models that use ontologies. Similarly, in the biomedical domain, huge ontologies (e.g., `http://obofoundry.org/`) are being developed and need to be integrated into decision making under uncertainty.

We adopt the approach of Poole, Smyth, and Sharma [2008], where ontologies form logical constraints which come logically prior to data and to probabilistic models. For example, as we know (as part of our ontology) that humans are mammals, we do not expect any dataset to say explicitly that some person is a human and a mammal (and if a person is said to be an animal, it is a different sense of animal). The data and the models are written taking into account the underlying ontology.

In an ontology, the domain of a property specifies the individuals for which the property is defined. Properties applied to individuals correspond to random variables in the probabilistic model. One problem in the integration of ontologies and reasoning under uncertainty arises when properties have non-trivial domains and thus the corresponding random variables are not always defined. For instance, the property *education* may be applicable to people but not to dogs or rocks. However, we may not know if an individual is a person, and performance on some task may depend on his/her education level if the individual is a person, such as when some damage may have been caused by a crafty person (depending on his/her education level) or by a natural phenomenon. Similarly, hardness measure (in Mohs scale) may be applicable to rocks but not to people. When modelling and learning, we do not want to think about undefined values, and we will never observe an undefined value in a dataset that obeys the ontology; e.g., we do not want to consider the education level of rocks, and no data would contain such information.

We first describe the result of Kuo et al. [2013], where probabilistic reasoning is carried out without using an undefined value by leveraging the logical dependencies in an ontology. Then we discuss the issues that arise when extending the modelling approach to the relational setting. For instance, suppose the property *friend* is defined for humans, and a *friend* of a human is another human. For different individuals, we may have varying beliefs about whether they are humans. A probabilistic model needs to take into account every pair of individuals that might be humans.

Another source of problems is when functional properties have classes (defining types of individuals) as their ranges, and we may be unsure about the types of individuals. When a functional property such as *bestFriend* has some population of individuals as its range, the probability that some particular person's *bestFriend* is a particular person may depend on th population size, about which we may be uncertain. Nevertheless, we want to build a probabilistic model that is useful for different populations of individuals. Probabilities must be specified in a way that varies with the population size for the range of the property.

For random variables whose ranges are parametrized by some population that is not known when constructing the model, we cannot specify actual individual probabilities, which would vary with the population size. We describe two

approaches to specifying these probabilities: (i) specify aggregate probabilities for groups of individuals and (ii) specify the relative proportions of individual probabilities.

Ontologies

An ontology is a formal specification of the meanings of the symbols in an information system [Smith, 2003]. Ontologies provide researchers and practitioners in a common community with standardized vocabulary in which to describe the world. An ontology defines any terminology that needs to be shared among datasets.

Modern ontologies, such as those in the Web Ontology Language (OWL) [Grau et al., 2008; Motik, Patel-Schneider, and Parsia, 2012], are defined in terms of classes, properties, and individuals. The semantics of OWL is defined in terms of sets [Motik, Patel-Schneider, and Grau, 2012] – a class is the set of all possible individuals in it, and a property is a binary relation R, which is the set of all (ordered) pairs for which the property holds. We write xRy if x is related to y by the property R (i.e., $\langle x, y \rangle \in R$).

Properties have domains and ranges. The domain of a property is the class of individuals for which the property is defined. Thus, a property is only defined for individuals in its domain; for other individuals, it is not defined. The range of a property is the set of possible values for the property. The range of a property may be a set of pre-specified values or some class of individuals.

Properties may be functional – each individual in the domain of a functional property is related to at most one value in its range (i.e., if xRy and xRy', then $y = y'$). OWL also defines other kinds of properties such as transitive, symmetric, and other meta properties.

It has been advocated that ontologies be written in terms of Aristotelian definitions [Aristotle, 350 BC; Berg, 1982; Poole, Smyth, and Sharma, 2009], where each explicitly specified class is defined in terms of a super-class, the *genus*, and restrictions on properties (e.g., that the property has a particular value), the *differentia*, that distinguish this class from other subclasses of the genus. This is not a restrictive assumption, as we can always define a property that is true of members of a class. It allows us to just model properties, with classes being defined in terms of property restrictions.

Thing is the single top-level class, of which every other class is a subclass, and it contains all possible individuals.

Example 1. Suppose *education* is a functional property only defined for humans, and its range is $\{low, high\}$. This can be specified in the OWL-2 functional syntax:

```
Declaration(ObjectProperty(:education))
FunctionalObjectProperty(:education)
ObjectPropertyDomain(:education :Human)
ObjectPropertyRange(:education
                ObjectUnionOf(:Low :High))
```

A human is an individual in the class *Animal* for which the property *isHuman* is true:

```
Declaration(Class(:Human))
EquivalentClasses(
  :Human
  ObjectIntersectionOf(
    DataHasValue(:isHuman "true"^^xsd:boolean)
```

```
    :Animal))
Declaration(DataProperty(:isHuman))
FunctionalDataProperty(:isHuman)
DataPropertyDomain(:isHuman :Animal)
DataPropertyRange(:isHuman xsd:boolean)
```

Here, we assume that *isHuman* is only defined for animals. An animal is an individual for which the property *isAnimal* is true and can be defined in a similar way. The domain of *isAnimal* is *Thing* and its range is a Boolean value. `xsd` in the above OWL-2 specification stands for the XML Schema Definition and refers to the schema where `boolean` is defined.

In Example 1, all the properties are functional, and their ranges are pre-specified sets of values. The genus and differentia of *Animal* are *Thing* and *isAnimal* = *true*, respectively, whereas those of the class *Human* are *Animal* and *isHuman* = *true*. The property *education* has *Human* as its domain, and is thus undefined when applied to an individual for which *isHuman* is not true. Likewise, *isHuman* has domain *Animal* and so is only defined for individuals of class *Animal*.

Example 2. The property *friend* is defined for humans, and the *friend* of a human is another human. Hence, both the domain and range of *friend* are *Human*:

```
Declaration(ObjectProperty(:friend))
ObjectPropertyDomain(:friend :Human)
ObjectPropertyRange(:friend :Human)
```

In Example 2, *friend* is a non-functional property whose range is a class of individuals. The elements of the set may vary; for example, the number of humans may be different in different problem settings. Example 3 gives the OWL-2 syntax for specifying a functional property with a class of individuals as its range.

Example 3. Similar to *friend*, the property *bestFriend* has *Human* as both its domain and range. However, a person can have at most one *bestFriend*.

```
Declaration(ObjectProperty(:bestFriend))
FunctionalObjectProperty(:bestFriend)
ObjectPropertyDomain(:bestFriend :Human)
ObjectPropertyRange(:bestFriend :Human)
```

Definition 1. An **ontology** O contains a set of logical assertions about a set C of classes, a set P of properties, and a set I of individuals. Each property $p \in P$ has a domain and a range. In addition, we assume O does not contain cyclic definitions of property domains in the sense that any class that defines the domain of a property cannot be defined in terms of that property.

An **Aristotelian ontology** is an ontology where every class $C \in C$ is either *Thing* or is defined as $C' \in C$ conjoined with a set of property restrictions (such as $p_1 = v_1 \wedge \cdots \wedge p_k = v_k$, where $p_i \in P$, $v_i \in range(p_i)$, and p_i is defined for individuals in C' for which $p_1 = v_1 \wedge \cdots \wedge p_{i-1} = v_{i-1}$). We can always reduce a class C to a conjunction of property restrictions, which we refer to as the **primitive form** of class C. Let $dom(p)$ be the primitive form of the domain of property p. The ontology induces a partial ordering of the properties: for every property $p \in P$, any property p' appearing

in $dom(p)$ precedes p, written as $p' \prec p$. Aristotelian definitions give rise to a class hierarchy and reduce class definitions to property restrictions.

The acyclic restriction avoids the possibility of defining two classes in terms of each other. It ensures that to determine whether a property is defined for an individual cannot involve applying the property to that individual.

A **formula**, i.e., logical sentence written in the language (e.g., OWL) used for an ontology, is entailed by the ontology if the formula is always true given the set of logical assertions in the ontology. As a simple example, if an ontology specifies that *Student* is a subclass of *Human* and *Human* is a subclass of *Animal*, then the ontology entails that *Student* is a subclass of *Animal* (i.e., any individual in the class *Student* is also in the class *Animal*). There exist ontology reasoners for OWL (e.g., Pellet[1] and HermiT[2]) that determine such entailments.

Ontologically-Based Probabilistic Models

Graphical models [Pearl, 1988; Lauritzen, 1996] represent factorizations of joint probability distributions in terms of graphs that encode conditional (in)dependencies amongst random variables. A belief network is a directed acyclic graphical model where the factorization represents conditional probability distributions (CPDs) of the random variables. A relational probabilistic model [Getoor and Taskar, 2007; De Raedt et al., 2008], also referred to as a template-based model [Koller and Friedman, 2009], extends a graphical model with the relational (or first-order) component. Such models can be defined in terms of parametrized random variables [Poole, 2003], which given a population of individuals, can be grounded into standard graphical models. A **parametrized random variable (PRV)** is of the form $F(t_1, \ldots, t_k)$, where F is a function symbol and each term t_i is either a logical variable associated with some population or a constant that corresponds to an individual. A parametrized random variable represents a collection of ground random variables, one for each assignment of an individual to a logical variable. Our work mainly concerns directed relational graphical models [Heckerman, Meek, and Koller, 2004; Kersting and De Raedt, 2007; Laskey, 2008] because the notion of definedness is inherently conditional.

We will build on the integration of ontologies and probabilistic models of Poole, Smyth, and Sharma [2009], where PRVs are constructed from properties and individuals. When modelling uncertainty, properties form PRVs, where the property becomes the function symbol of the PRV.

A non-functional property R is modelled as a PRV of the form $R(X, Y)$ with two arguments. When we are unsure about the types of individuals, the grounding contains all pairs of individuals which are possibly in the domain and range of R, and the range of the PRV is $\{true, false, \bot\}$, where $\bot$ represents *undefined*. We call a (parametrized) random variable with such range an extended Boolean (parametrized) random variable. The main result of Kuo et al. [2013] is how (for the non-relational case) to do inference

when random variables are only defined in some contexts without explicitly reasoning about the value *undefined*.

Example 4. The non-functional property *friend* in Examples 2 has a range whose size can vary. We construct a PRV $friend(X, Y)$, where both X and Y are logical variables. The population for the grounding is all of the individuals who could possibly be human.

A functional property with a fixed range becomes a multi-valued PRV of the form $R(X)$, which is parametrized by the domain of the property, and the range of the PRV is the range of the property extended with $\bot$.

Example 5. To model the property *education* in Example 1, we construct a parametrized random variable $education(X)$, whose range is the range of the property extended with $\bot$, i.e., $\{high, low, \bot\}$.

It is more complicated to model functional properties whose ranges are classes of individuals which are not fixed. A functional property whose range may vary is still modelled as a parametrized random variable of the form $R(X)$. However, in addition to being parametrized by the logical variable X, the range of such random variable is also implicitly parametrized by the population associated with the range of the corresponding property.

Example 6. Consider modelling the functional property *bestFriend* in Example 3. We construct a parametrized random variable $bestFriend(X)$. The range of $bestFriend(X)$ includes the set of all humans and so is also parametrized by the population of *Human*.

Functional properties add an additional layer of complexity over non-functional properties. If we had chosen to treat $bestFriend(X, Y)$ as an extended Boolean PRV similar to $friend(X, Y)$, then the ground instances $bestFriend(x, y_1), bestFriend(x, y_2), \ldots$ would have to be fully connected because only one of them can be true. We can arrange PRVs into a DAG, which specifies (conditional) dependencies, forming a relational probabilistic model defined in terms of PRVs.

Modelling Undefined Properties

One of the problems in building (relational) probabilistic models with an underlying ontology is that the property associated with a random variable is undefined for individuals not in its domain. A random variable is only defined in the contexts when its corresponding property is applied to an individual in the domain of that property. When building a model, we only want to use a random variable in a context where the random variable is defined. As Poole, Smyth, and Sharma [2009] noticed, the idea of using a random variable only in a context in which it is defined is reminiscent of context-specific independence [Boutilier et al., 1996; Poole and Zhang, 2003]. Instead of context-specific independence, here a random variable is simply not defined in some contexts. We need a way of representing and reasoning with potentially undefined random variables.

A straightforward approach is to simply add an extra value "$\bot$" to the range of each random variable. However, when combined with the restriction that a random variable is only

used in a context where it is defined, Kuo et al. [2013] propose to separate ontological dependencies from probabilistic dependencies in an ontologically-based probabilistic model, showing how to avoid explicit modelling of undefined values in the random variables.

The central idea is straightforward. We avoid probabilistic dependencies of a random variable on those random variables that can render it undefined. When querying the posterior distribution of a random variable Q, we first query whether the ontology entails that the observed evidence implies the definedness of Q, and, if not, we then separately query (i) the probability that Q is defined and (ii) the distribution of Q as if we knew it was defined. Neither of these two queries uses the extra value $\perp$. In particular, if the query variable Q is defined, we do not need to care about whether the other random variables are defined during inference. By exploiting the determinism in the ontology, we can avoid modelling "$\perp$" the in the range of the random variables and also largely reduce the probabilistic network structure. Probabilistic reasoning can thus be greatly sped up.

In the following sections, we discuss some of the issues that arise when extending this modelling approach to the relational setting.

Type Uncertainty about Individuals

In an ontologically-based probabilistic model, random variables are parametrized over populations, each of which represents some class (i.e., type) of individuals. Most work in combining logic and probability addresses situations where the types of the individuals in a population are known, but the population sizes or individual identities may be uncertain. When modelling real-world scenarios, we often have type uncertainty about the individuals [Koller, Levy, and Pfeffer, 1997; Poole and Smyth, 2005; Poole, 2007]; it is common that we have different levels of knowledge about the types of different individuals. Such type uncertainty needs to be taken into consideration when reasoning.

Example 7. We may have a population of 100 individuals and know 80 of them are animals, 30 of these animals are humans, but have no knowledge about the type(s) of the remaining 20 individuals. Each of the 50 individuals, for whom all we know is that they are animals, could either be human or non-human. Similarly, the 20 individual whose types are unknown to us could be animals or non-animals.

A non-functional property whose range is some class gives rise to a random variable parametrized by both the domain and range of the property. When we do not know the number of individuals in the class for the range of a property, we may not even know the number of ground instances of the parametrized random variable we need to reason about.

Example 8. Let *friendliness* be a functional property whose domain is *Human* and range is {*true*,*false*}. Consider the property *friend* in Example 2, and suppose that whether two people are friends depends on the *education* and *friendliness* of both of them. Figure 1 depicts the part of a relational probabilistic model that specifies the dependency of *friend* on other properties, where these properties correspond to parametrized random variables in the model.
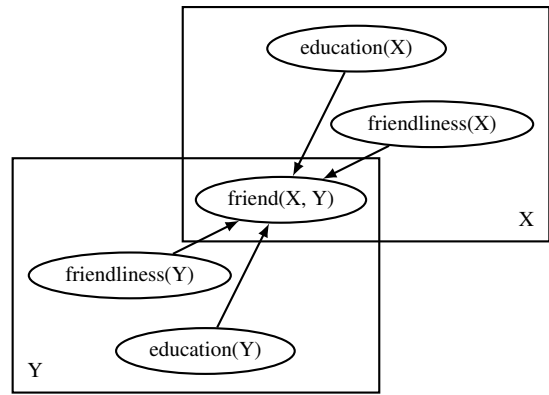

Figure 1: Part of an ontologically-based relational model containing the extended Boolean PRV *friend*(X,Y) and its dependency on other variables. The plates indicate replicates for each instantiation of the logical variable(s).

The result from Kuo et al. [2013] suggests that we could, in principle, consider every pair of individuals (even though we may be uncertain about whether properties *education*, *friendliness*, or *friend* are defined for the individuals), as long as the model in Example 8 specifies the probability that an individual is a human. This approach can be very inefficient when there are a vast number of individuals, and some individuals are known to be non-human from some other properties. When reasoning, we want to only consider the ground instances of the PRVs for which the pair of individuals have a non-zero probability of being humans.

Functional Properties with Varying Ranges

Modelling and reasoning become more complicated when we consider functional properties whose ranges are classes of individuals, as the ranges of the corresponding (parametrized) random variables vary with the population.

Example 9. Consider the PRV *bestFriend*(X) in Example 6. Suppose that, for two humans x and y, whether the *bestFriend* of x is y depends on the *education* of both x and y, as well as the number of lowly and highly humans in the domain. (For notational brevity, *friendliness*(X) and *friendliness*(Y) are dropped for this example.) See Figure 2 for a graphical model representation.

In Example 9, *bestFriend*(X) is parametrized by Y in a way different from the parametrization by X. For any individual x, the possible values in the range of *bestFriend*(x) includes the individuals that can fill the role of Y. This parametrization of the range of a random variable is, in some sense, similar to the latent Dirichlet allocation [Blei, Ng, and Jordan, 2003] for topic modelling, where the topic variable's range would be parametrized by the topics when the set of topics were not known a priori.

Instead, we could have chosen to model *bestFriend*(X,Y) as an extended Boolean PRV. The graphical model representation would look very similar to that shown in Figure 1. However, we would need a way to enforce the constraint that
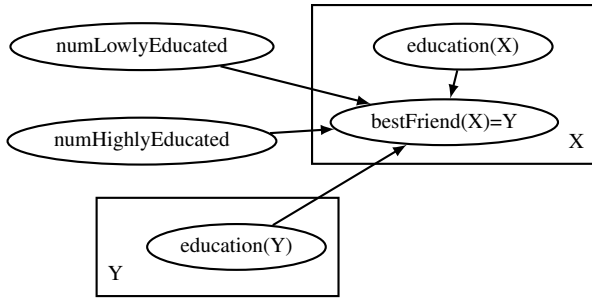
Figure 2: Part of an ontologically-based relational model specifying $bestFriend(X)$'s dependency on other variables.

for any instantiation x, there is at most one individual y such that $bestFriend(x,y)$ is true. This constraint requires that all ground instances $bestFriend(x,y_1), bestFriend(x,y_2)\ldots$ be fully connected and is not easily captured in the graphical model representation.

In Example 9, the parents of $bestFriend(X)$ are parametrized by logical variables that are associated with the range of $bestFriend(X)$. As such, the probabilities that $bestFriend(X) = y$, for some given human y, depends not only on $education(y)$, but also on the number of individuals that can fill the role of Y. We describe two different approaches to specifying probabilities in such situation.

In the first approach, conditional probabilities are specified in a usual way with a probability for each combination of assignments to the parent variables such that these probabilities sum to 1. However, these probabilities are aggregate probability masses for groups of individuals. When reasoning about any single individual, we need to uniformly distribute these probability masses to the individuals in the population, as these individuals are indistinguishable a priori (if we are not given specific information about any particular individual).

Example 10. Following Example 9, we have a model that specifies that for $X \neq Y$,

$$P_{\Sigma_Y}(bestFriend(X) = Y \mid$$
$$education(X) = high, education(Y) = high) = 0.36,$$

and similarly for each of the other joint assignments to $education(X)$ and $education(Y)$ such that these numbers sum to 1. The subscript Σ_Y indicates that the specified number is actually the sum of probabilities for all the instances of Y.

The number 0.36 represents an aggregate probability – for some highly educated human, the probability that his $bestFriend$ is a highly educated human is 0.36. To obtain individual probabilities, we divide the aggregate probability by the number of highly educated humans. Suppose that Chris is a highly educated human and that the population contains 100 highly educated humans. The probability that Chris' $bestFriend$ is some other highly educated human (e.g., Mary), for whom we have no specific beliefs, is $\frac{0.36}{99} = 0.00\overline{36}$.

In the case where we do not know the number of individuals in the class for the range of the property (i.e., when

there is type uncertainty about individuals), we need to calculate the expected individual probability over the number of individuals to obtain the probability for any single individual. This, however, may result in values that do not have a closed form. One heuristic is to simply divide the probability masses by the expected number of individuals in the class.

Example 11. Continuing Example 10, but assume we do not know the number of humans. Let there be 100 individuals. Suppose the model specifies that an individual is an animal with probability 0.9, that an animal is a human with probability 0.7, and that a human is highly educated with probability 0.3. Then the probability that an individual is a highly educated human is $q = 0.9 \times 0.7 \times 0.3$. The expected size of the animal population is 90, that of the human population is 72, and the expected number of highly educated humans is 21.6.

The probability that Chris' $bestFriend$ is Mary is calculated as an expectation over the population size of highly educated humans:

$$\frac{1}{Z} \sum_{i=2}^{100} \frac{0.36}{i} \binom{100}{i} q^i (1-q)^{100-i} \approx 0.01996$$

where $Z = \sum_{j=2}^{100} \binom{100}{j} q^j (1-q)^{100-j} = 1 - \sum_{j=0}^{1} \binom{100}{j} q^j (1-q)^{100-j}$ is the normalizing term, and we have assumed that the aggregate probability is positive implies there are at least two highly educated humans. A heuristic to calculate the probability that Chris' $bestFriend$ is Mary can be $\frac{0.36}{20.6} \approx 0.01748$.

If we have different beliefs for certain specific individuals, we need to adjust these probabilities so that they add up to the remaining probability mass.

An alternative and what we believe more sensible approach to is use weights, similar to those used for undirected graphical models. Instead of aggregate probability mass, for each combination of assignments to the parent variables, we specify a weight that represents the relative portion of an individual probability.

Example 12. Consider Example 9 again. Our model now specifies, for $X \neq Y$, the following weights:

$$\phi(bestFriend(X) = Y \mid$$
$$education(X) = high, education(Y) = high) = 3$$
and
$$\phi(bestFriend(X) = Y \mid$$
$$education(X) = high, education(Y) = low) = 1$$

as well as a weight 2 for any other possible combination of assignments to the parent variables.

A key difference between these weights and those used for undirected models is that the weights here can be interpreted locally. They give the odds ratio between probabilities for single individuals in different groups. In Example 12, the first two weights indicate that for two highly educated humans x and y_1 and some lowly educated human

y_2, *bestFriend*(x) is 3 times more likely to be y_1 than y_2. Given the relevant population sizes or the distributions of those population sizes, individual conditional probabilities can be calculated locally.

Example 13. Continuing Example 12, suppose that the population size is 50 for highly educated humans and 500 for lowly educated humans. We can calculate the probability that Chris' *bestFriend* is Mary as

$$\frac{3}{2+3 \times (50-1) + 1 \times 500} \approx 0.00462.$$

Compared with the first approach, which specifies aggregate probabilities at the population level, this alternative specifies weights that contribute at the individual level. The aggregate probability for a particular group (e.g., highly educated) of humans will change with the (distribution of) group size. In most scenarios, we believe this is the more realistic modelling approach.

Symmetric and Transitive Properties

In this section, we briefly mention some other kinds of properties in an ontology, how we can model them as (parametrized) random variables in an ontologically-based probabilistic model, and what issues they may raise.

A symmetric property represents a binary relation whose elements are unordered pairs. Formally, property R is symmetric if it is true that xRy if and only if yRx (i.e., $\langle x,y \rangle \in R \Leftrightarrow \langle y,x \rangle \in R$).

Example 14. Suppose the non-functional property *fbFriend* represents a friendship relation on Facebook. To establish such relationship requires confirmations from both people, so that each person agrees the other is his/her friend. Therefore, *fbFriend* is symmetric.

Unlike for *friend*, the set of ground instances represented by the PRV *fbFriend*(X,Y), constructed from the property *fbFriend* in Example 14, should include only one instance for each unordered pair of individuals. For two individuals h_1 and h_2, we want to only reason about one of the two ground random variables, *fbFriend*(h_1,h_2) and *fbFriend*(h_2,h_1).

A transitive property R satisfies that if xRy and yRz, then xRz (i.e., $\langle x,y \rangle \in R \wedge \langle y,z \rangle \in R \Rightarrow \langle x,z \rangle \in R$).

Example 15. Let *above* be a non-functional property whose domain and range are *Block*. For two blocks X and Y, *above*(X,Y) means Y is above X. It is natural (and realistic) to model this *above* relationship as a transitive property.

To model the property *above* in Example 15 as an extended Boolean PRV *above*(X,Y), we need to make sure that transitivity holds amongst all ground instances. To do so, however, will make the ground instances densely connected with each other in the probabilistic model.

One approach to modelling a transitive property R is to model another property R' such that R is the transitive closure of R'. In Example 15, instead of using *above*, we could model the property *immediatelyAbove*. By doing so, we remove the issue of transitivity, and *immediatelyAbove* is just a non-functional property as discussed before.

Conclusion

This paper discusses some of the issues that pertain to integrating ontologies with reasoning under uncertainty. In particular, it concerns building relational probabilistic models that use properties from an ontology. Different kinds of properties in an ontology have been considered, and we have described how we might construct parametrized random variables from these properties to build a relational probabilistic model.

We also described and explained two different approaches to specifying conditional probabilities for random variables whose range is a class of individuals such that the population may vary. Certain related problems have also been identified and discussed.

Acknowledgement

References

Aristotle. 350 B.C. *Categories*. Translated by E. M. Edghill.

Berg, J. 1982. Aristotle's theory of definition. In *ATTI del Convegno Internazionale di Storia della Logica*, 19–30.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research* 3:993–1022.

Boutilier, C.; Friedman, N.; Goldszmidt, M.; and Koller, D. 1996. Context-specific independence in Bayesian networks. In *Proceedings of the Twelfth Conference on Uncertaintyin Artificial Intelligence (UAI-96)*, 115–123.

De Raedt, L.; Frasconi, P.; Kersting, K.; and Muggleton, S. H., eds. 2008. *Probabilistic Inductive Logic Programming*. Springer.

Getoor, L., and Taskar, B., eds. 2007. *Introduction to Statistical Relational Learning*. Cambridge, MA: MIT Press.

Grau, B. C.; Horrocks, I.; Motik, B.; Parsia, B.; Patel-Schneider, P.; and Sattler, U. 2008. Owl 2: The next step for owl. *Web Semant.* 6(4):309–322.

Heckerman, D.; Meek, C.; and Koller, D. 2004. Probabilistic models for relational data. Technical Report MSR-TR-2004-30, Microsoft Research.

Kersting, K., and De Raedt, L. 2007. Bayesian logic programming: Theory and tool. In Getoor, L., and Taskar, B., eds., *An Introduction to Statistical Relational Learning*. MIT Press.

Koller, D., and Friedman, N. 2009. *Probabilistic Graphical Models - Principles and Techniques*. MIT Press.

Koller, D.; Levy, A.; and Pfeffer, A. 1997. P-Classic: A tractable probabilistic description logic. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97)*, 390–397. AAAI Press.

Kuo, C.-L.; Buchman, D.; Katiyar, A.; and Poole, D. 2013. Probabilistic reasoning with undefined properties

in ontologically-based belief networks. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI-13)*. To appear.

Laskey, K. B. 2008. MEBN: A language for first-order Bayesian knowledge bases. *Artificial Intelligence* 172(2–3):140–178.

Lauritzen, S. L. 1996. *Graphical models*, volume 17. Oxford University Press.

Motik, B.; Patel-Schneider, P. F.; and Grau, B. C. 2012. Owl 2 web ontology language: Direct semantics (second edition). *W3C Recommendation, 11 December 2012*.

Motik, B.; Patel-Schneider, P. F.; and Parsia, B. 2012. Owl 2 web ontology language: Structural specification and functional-style syntax (second edition). *W3C Recommendation, 11 December 2012*.

Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufmann.

Poole, D., and Smyth, C. 2005. Type uncertainty in ontologically-grounded qualitative probabilistic matching. In *Proc. European Conference on Symbolic and Qualitative Reasoning About Uncertainty (ECSQARU)*, LNAI 3571, 763–774. Barcelona, Spain: Springer-Verlag.

Poole, D., and Zhang, N. L. 2003. Exploiting contextual independence in probabilistic inference. *Journal of Artificial Intelligence Research* 18:263–313. Code is available from `http://www.cs.ubc.ca/~poole/code.html`.

Poole, D.; Smyth, C.; and Sharma, R. 2008. Semantic science: Ontologies, data and probabilistic theories. In *Uncertainty Reasoning for the Semantic Web I*, LNAI/LNCS. Springer.

Poole, D.; Smyth, C.; and Sharma, R. 2009. Ontology design for scientific theories that make probabilistic predictions. *IEEE Intelligent Systems* 24(1):27–36.

Poole, D. 2003. First-order probabilistic inference. In *Proceedings of the Eighteenth International Joint Conference on Aritficial Intelligence (IJCAI-03)*, 985–991.

Poole, D. 2007. Logical generative models for probabilistic reasoning about existence, roles and identity. In *Proceedings of the Twenty-Second National Conference on Artificial Intelligence (AAAI-07)*, 1271–1277.

Smith, B. 2003. Ontology. In Floridi, L., ed., *Blackwell Guide to the Philosophy of Computing and Information*. Oxford: Blackwell. 155–166.