A System for Ontologically-Grounded Probabilistic Matching

Rita Sharma
Georeference Online Ltd. and
Dept. of Computer Science, UBC
http://www.cs.ubc.ca/∼rsharma/

David Poole
Department of Computer Science
University of British Columbia
http://www.cs.ubc.ca/∼poole/

Clinton Smyth
Georeference Online Ltd.,
http://www.georeferenceonline.com

Abstract

This paper is part of a project to match descriptions of real-world instances and probabilistic models, both of which can be described at multiple level of abstraction and detail. We use an ontology to control the vocabulary of the application domain. This paper describes the issues involved in probabilistic matching of hierarchical description of models and instances using Bayesian decision theory, which combines ontologies and probabilities. We have two fielded applications of this framework; one for landslide prediction and one for mineral exploration.

1 Introduction

In many problem domains we need to match instances and models of real-world phenomena. For example, in geology, geological surveys of states, provinces and countries publish descriptions of mineral occurrences in their jurisdiction; these form the instances in one of our applications. People spend careers describing probabilistic models of where different minerals can be found. There are two main tasks we consider:

- given an instance, determine which models best fits it. This would be used, for example, by someone who has the mineral rights on a piece of land and wants to know what mineral deposits may be there based on the description of the property.
- given a model, determine which instances best match the model. This would be used by someone who has a model of where gold can be found, and they want to find which piece of land is most likely to contain gold, based on their model.

These models and instances are typically described by different people at different levels of abstraction (some use more general terms than others) and different levels of detail (some have parts and sub-parts and some may be described holistically). Descriptions of mineral occurrences are recorded at varied levels of abstraction and detail because some areas have been explored in more detail than others. There are some models that people spend careers in developing and that are described in great detail for those parts that the modeler cares about. Other models are less well developed, and described only in general terms. Because the instance and model descriptions are generated asynchronously, the levels of detail cannot be expected to match. We do, however, need to make decisions based on all of the information available.

This work has arisen from from an ongoing project in which we are building decision-making tools for mineral exploration (MineMatch) and hazard mapping (HazardMatch). MineMatch is similar in its goals to the Prospector expert system [Hart, 1975], but builds on the developments in probabilistic reasoning and ontologies of the last 30 years. In previous work [Smyth and Poole, 2004; Poole and Smyth, 2005], we described models using qualitative probabilities, based on the kappa calculus, which measures uncertainty in degree of "surprise". In this paper, we develop an approach based on probability for making decisions.

In MineMatch we work with more than 25,000 instances of mineral occurrences that are described using various taxonomies, including the British Geological Survey Rock Classification scheme[1] and the Micronex taxonomy of Minerals[2]. We also work with more than 100 deposit type models, including those described by the US Geological Survey[3] and the British Columbia Geological Survey[4]. Similarly, in HazardMatch we work with tens of thousands of spatial instances (polygons) described using standard taxonomies of environmental modeling such as rock type, geomorphology and geological age. To date we

[1] http://www.bgs.ac.uk/bgsrcs/
[2] http://micronex.golinfo.com
[3] http://minerals.cr.usgs.gov/team/depmod.html
[4] http://www.em.gov.bc.ca/Mining/Geolsurv/

have worked with approximately ten models of landslide hazards which we compare with the spatial instances.

This work is quite different to other work on combining probability and ontologies [Ding and Peng, 2004; Pool, Fung, Cannon and Aikin, 2005; Costa, Laskey and Laskey, 2005] because we are using the ontologies to construct a rich hypotheses space rather than (only) having probabilities over the ontologies. The running example we use in this paper is one where we can describe apartments and/or houses and their models.

2 Models and Instances

Instances are things in the world. We describe instances by naming them and specifying their features (values on various properties). For example, an instance could be a particular rock outcrop, a volcano that used to exist, or apartment #103 at 555 Short St. A feature of that apartment could be that its size is large and it contains two bathrooms.

Models are concepts in someone's head that describe some phenomenon of interest. For example, someone may have a model of what rocks are likely to contain gold, a model of where landslides may occur, or a model of an apartment that Sue would be content to live in. In the system we consider here, models are named and described in terms of probability distributions over the features of an instance that manifests that phenomenon. For example, the gold model will specify the probability over the features of a particular instance that is likely to contain gold. The landslide model will specify the probability over the features for a particular location that predict whether that location is prone to landslides. The model of Sue's apartment will specify the features that predict whether Sue would be expected to like a particular apartment.

Given an instance and a model, the aim of matching, in the context of this paper, is to determine the probability that the instance manifests the phenomenon of the model.

3 Ontologies

The models and instances are described at different levels of abstraction using ontologies. As part of the ontologies we assume that we have taxonomic hierarchies that specify the vocabulary for different levels of abstraction. The taxonomic hierarchy defines the hierarchical relationship between concepts. Figure 1 shows an example of a taxonomic hierarchy. A *bedroom* is a kind of *room*. A *masterbedroom* is a kind of *bedroom*. In this figure, *room* is the topmost class.

We do not assume that the ontologies include uncertainty about properties and relations. Ontologies are created and maintained by communities, which can agree on vocabulary, even if they do not agree on probabilities and models.

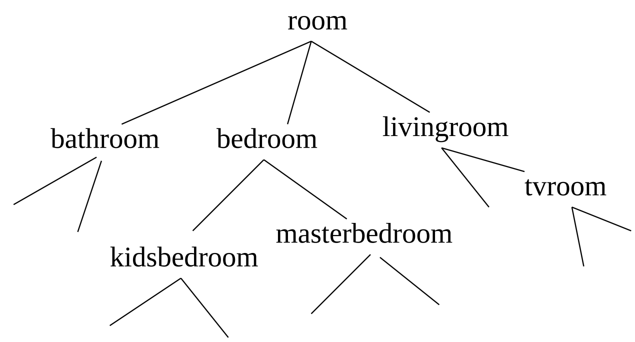

Figure 1: Part of a taxonomic hierarchy of room types.

The ontologies provide a hypothesis space over which we can have probability distribution. We consider that probabilistic models (scientific theories) that makes probabilistic prediction about a domain will provide the uncertainty knowledge about properties and relations.

4 Describing Model and Instances

We adopt the OWL [McGuinness and van Harmelen, 2004] terminology of describing domains in terms of individuals, classes and properties.

4.1 Instances

An instance is described by its value on various properties. This can include its relationship to other individuals (e.g., its parts). We, however, do not only want to state positive facts, but also negative facts such as that an apartment does not contain a bedroom, or that the kitchen is a red colour but is not a pink (without enumerating all of the non-pink red colours). Thus we will represent instance descriptions with the quadruples of form:

$$\langle individual, property, value, truthvalue \rangle$$

where *truthvalue* is either *present* or *absent*

For example, to say that an apartment has a master bedroom, but does not have a kid's bedroom we could write:

$$\langle apt1, containRoom, masterbedroom, present \rangle$$

$$\langle apt1, containRoom, kidsbedroom, absent \rangle$$

It is important to distinguish an instance from its description. An instance is a real physical thing that exists in the world we are reasoning about (the real world at some time, some temporally extended world, or even some imaginary world). A description is a set of quadruples.

4.2 Models

Models describe abstract instances rather than any particular instance. For example, apartment model Apt_13 may

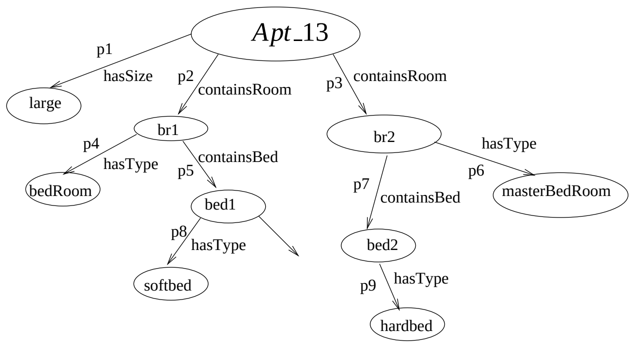

Figure 2: A Semantic network representation of apartment model Apt_13.

describe features that Sue would love to have in an apartment, e.g., she usually wants a master bedroom in her apartment[5]. In particular, a model describes an instance that exhibits some phenomenon. It specifies what must be in an instance, what cannot be there and what we would expect to be there.

A model describes a set of related individuals. One of these individuals is the designated top-level individual. For example, in model Apt_13 the individuals are the apartment, bedrooms, beds etc, and the designated top-level individual is the apartment.

A model is described in terms of quadruples of the form:

$$\langle ind, prop, val, prob \rangle$$

where *ind* is an individual, *prop* is a property, *val* is either an individual or a class or a primitive value (depending on whether *prop* is an objecttype property, or a *hasType* property, or a datatype property), and *prob* is the probability.

This quadruple specifies that an instance individual that has value *val* on property *prop* will matches the model individual *ind* with probability *prop*.

Example The semantic network representation of part of an apartment model Apt_13 is shown in Figure 2. The nodes represent individuals, classes and data types. The *top object* in a semantic network represents the individual that we are talking about. The individual Apt_13 in Figure 2 is an apartment. Each arc is labeled with a probability. The value *val* associated with probability *prob*, individual *ind*, and arc from *ind* to *val*, labeled with property *prop*, represent quadruple $\langle ind, prop, val, prob \rangle$. For example, individuals *Apt_13*, *br1* and the arc connecting these two individuals represent quadruple $\langle Apt\_13, containsRoom, br1, p2 \rangle$.

[5]For this paper do not think of these as preferences. We could have a similar matcher for preferences, but this paper is about models of uncertainty. Think of the model of what Sue would like as the probability that she will move into the apartment and still be there after 6 months. This is, in fact, what the landlord is interested in.

5 Abstraction hierarchies and probabilities

When matching a model with an instance, we need to take into consideration the type uncertainty (because the instance and model are at varied levels of abstraction). To cope with type uncertainty, we consider that taxonomic hierarchies in the ontology are associated with probabilities. In particular, given a taxonomic hierarchy, we want a mechanism that can compute $P(C_j|C_k)$, where C_j is the *subClassOf* C_k. This is the probability that an individual is a C_j given all that you know about it is that it is a C_k.

We are not considering that the probabilities associated with hierarchies are part of individual models. We are considering them as a part of *super model*.

In this paper we consider only taxonomic hierarchies which are trees and where we can compute $P(C_j|C_k)$, as discussed in Section 5.1. We are working on techniques for computing $P(C_j|C_k)$, when hierarchies are not trees, and where we need to consider the problem of multiple inheritance, and interdependence between subclasses.

5.1 Tree abstraction hierarchies

Each class in a taxonomic hierarchy specifies a probability distribution over its immediate subclasses. That is, each link in the tree hierarchy is associated with a conditional probability. This is the probability that an individual is in a class C_j, given that all you know about it is that it is in a class C_k, and that C_j is the immediate subClassOf C_k. For example, the class *room* in the hierarchy shown in Figure 1 has a probability distribution over its immediate subclasses. Suppose we have as part of this distribution:

$$P(bedroom|room) = 0.3$$

$P(bedroom|room)$ represents the probability that a random room is a *bedroom*.

Similarly, we can specify the probability of an immediate subClassOf *bedroom* given *bedroom*, with probabilities such as:

$$P(masterbedroom|bedroom) = 0.2$$

$P(masterbedroom|bedroom)$ represents the probability that a room is a *masterbedroom* given all that you know about it is that it is a *bedroom*.

The prior probability that an individual is in a class can be computed in a recursive manner by multiplying the probabilities up in the tree. The probability that an individual belongs to root class (*room*) is 1 (as it represents the set of all individuals). That is, $P(room) = 1$. For example, given the probability as above, $P(masterbedroom)$ can be calculated as follows:

$$\begin{aligned} & P(masterbedroom) \\ &= P(masterbedroom|bedroom) \times \\ & \quad P(bedroom|room) \times P(room) \\ &= 0.2 \times 0.3 \end{aligned}$$

In this representation, computing the probability that $i \in C_k$ given that $i \in C_j$ is linear in depth difference of C_j and C_k and otherwise is not a function of the hierarchy's size.

6 Supermodel

As discussed in Section 4.2 a model describes a concrete instance that matches that model. In particular, it specifies what must be in an instance, what cannot be there and what we would expect to be there. However, a model does not specify what happens when the model doesn't hold (as that depends on what other models there are, and the background probabilities). The role of the supermodel is to provide background information (that is beyond any model) on how likely *individual—property—value* triples are. In particular, the super model contains the following:

- the supermodel contains the probability distribution of each class in the tree abstraction hierarchies as discussed in Section 5.1.

- the supermodel contains quadruples of the form:

  $$\langle cl, prop, val, prior \rangle$$

  where *cl* is a class in the taxonomic hierarchy, *prop* is a property, *val* is either an individual or a class or a primitive value, and *prior* is the prior (background) probability.

  That is, the prior probability that an individual of type *cl* has value *val* for property *prop* is available from the supermodel. For example, quadruple $\langle room, hasColour, \text{“}green'', 0.4 \rangle$ tells us that the prior probability of a random room has “green” colour is 0.4.

7 Probabilistic Matching

One objective of the matcher is to rank the models or instances given instance and model descriptions. The basic problem is to match an instance with a model. When we say that a model M matches an instance i, we write $M \sim i$ to mean that M matches with i. Note that M is the top-level individual in the model and i is the top-level individual in the instance. We want to determine the posterior probability of $M \sim i$ given the i's description, which specifies the probability that the instance i manifests the phenomenon that the model is modeling.

In general, $M_k \sim i_j$ represents that model individual M_k matches the instance individual i_j, where a model individual is one of the individuals described in the model (i.e., it is one of the first elements of a quadruple), and an instance individual is one of the individuals described in the instance description.

We cannot directly determine the match between model and instance unless we know which model individuals correspond to which instance individuals.

We use $M_k = i_j$ to denote that model individual M_k corresponds to instance individual i_j and $M_k =\perp$ to denote that individual M_k does not corresponds to any instance individual. A role assignment is a list of correspondence statements of the forms $M_k = i_j$, $M_k =\perp$ such that each M_k appears exactly once in the list and each i_j appears at most once.

Note that match, $\sim$, does not define the role assignment. It defines the degree of match, given a role assignment.

Given a role assignment, the model description defines a Bayesian network. The problem of matching a model M with an instance i reduces to computing $P(M \sim i|observation)$ from the constructed Bayesian network, where observation is the instance i's description.

7.1 Construction of Bayesian network

Given a role assignment, the semantic network defines a Bayesian network. We can construct it dynamically during the inference as follows:

- there is a Boolean node $\langle M_k \sim i_j \rangle$ for each correspondence statement $M_k = i_j$, where $i_j \neq \perp$ of the role assignment.

- there is a Boolean node for each correspondence statement $M_k =\perp$ of the role assignment, which we will write $\langle M_k =\perp \rangle$. This node will be observed with value *true*.

- for each individual M_k in the model description and for each functional property *prop* such that *prop* is *hasType* or datatype, there is a random variable which we will write $\langle M_k, prop \rangle$. The domain of $\langle M_k, prop \rangle$ is the range of property *prop*.

- for each individual M_k in the model description and for each non-functional property *prop* such that *prop* is datatype or the range of *prop* is class (i.e., *prop* is *hasType*) and for each value V in the range of *prop*, there is a Boolean variable, which we will write $\langle M_k, prop, V \rangle$.

- the parent of each $\langle M_k \sim i_j \rangle$ node, and each $\langle M_k = \perp \rangle$ node, is node $\langle M_p \sim i_p \rangle$ such that there is a directed edge from M_p to M_k in the semantic network (i.e., quadruple $\langle M_p, prop, M_k, prob \rangle$ exists in the model description).

- the parent of each $\langle M_k, P \rangle$, and each $\langle M_k, P, V \rangle$ node is node $\langle M_k \sim i_j \rangle$.

- the probability distribution of each $\langle M_k \sim i_j \rangle$ node conditioned on its parent $\langle M_p \sim i_p \rangle$ is:

$$\begin{aligned} P(\langle M_k \sim i_j \rangle = true | \langle M_p \sim i_p \rangle = true) &= p \\ P(\langle M_k \sim i_j \rangle = true | \langle M_p \sim i_p \rangle = false) &= prior \end{aligned}$$

  where p is the probability associated with the individual M_k in the semantic network. That is, quadruple $\langle M_p, prop, M_k, prob \rangle$ is the part of the model description. The prior probability *prior* is taken from quadruple $\langle M_p, prop, M_k, prior \rangle$ that exists in the supermodel.

- the probability distribution for each $\langle M_k = \perp \rangle$ node conditioned on its parent $\langle M_p \sim i_p \rangle$ is:

$$\begin{aligned} P(\langle M_k = \perp \rangle = true | \langle M_p \sim i_p \rangle = true) &= 1 - p \\ P(\langle M_k = \perp \rangle = true | \langle M_p \sim i_p \rangle = false) &= 1 - prior \end{aligned}$$

  where p is the probability associated with the individual M_k in the semantic network. That is, quadruple $\langle M_p, prop, M_k, p \rangle$ is the part of model description. The prior probability *prior* is taken from quadruple $\langle M_p, prop, M_k, prior \rangle$ that exists in the supermodel.

- The domain of a $\langle M_k, prop \rangle$ node is the range of *prop*. To specify the conditional probability of $\langle M_k, prop \rangle$ node conditioned on its parent $\langle M_k \sim i_j \rangle$, we do not have the distribution over all the values of $\langle M_k, prop \rangle$, rather, we have the probability for values that model cares about. The conditional probability $P(\langle M_k, prop \rangle \,|\, \langle M_k \sim i_j \rangle$ is:

  - If the range of *prop* is class, $P(\langle M_k, prop \rangle \,|\, \langle M_k \sim i_j \rangle)$ is:

$$\begin{aligned} P(\langle M_k, prop \rangle \in V | \langle M_k \sim i_j \rangle = true) &= prob \\ P(\langle M_k, prop \rangle \in V | \langle M_k \sim i_j \rangle = false) &= prior \end{aligned}$$

  - If *prop* is datatype property:

$$\begin{aligned} P(\langle M_k, prop \rangle = V | \langle M_k \sim i_j \rangle = true) &= prob \\ P(\langle M_k, prop \rangle = V | \langle M_k \sim i_j \rangle = false) &= prior \end{aligned}$$

  where *prob* is the probability associated with quadruple $\langle M_k, prop, V, prob \rangle$ in the model description. The prior probability *prior* is taken from the quadruple $\langle M_k, prop, V, prior \rangle$ that exists in the supermodel.

- The probability distribution for each $\langle M_k, prop, V \rangle$ node conditioned on its parent $\langle M_k \sim i_j \rangle$ is:

$$\begin{aligned} P(\langle M_k, prop, V \rangle = true | \langle M_k \sim i_j \rangle = true) &= p \\ P(\langle M_k, prop, V \rangle = true | \langle M_k \sim i_j \rangle = false) &= prior \end{aligned}$$

  where p is the probability associated with value V in the semantic network. That is, quadruple $\langle M_k, prop, V, p \rangle$ exists in the model description. The prior probability *prior* is defined by the supermodel, i.e., quadruple $\langle M_k, prop, V, prior \rangle$ exists in the supermodel.

Example Consider matching the apartment model Apt_13 as shown in Figure 2 with instance $apt1$ defined as follows:

$\langle apt1, hasSize, \text{“}large''\rangle$
$\langle apt1, containsRoom, R1, present \rangle$
$\langle R1, type, masterbedroom, present \rangle$
$\langle R1, containsBed, b1, present \rangle$
$\langle b1, type, bed, present \rangle$
$\langle apt1, containsRoom, R2, present \rangle$
$\langle R2, type, room, present \rangle$

For the individual $br1$ of the model as shown in Figure2, we can have the following possible mappings:

$br1 = R1$
$br1 = R2$
$br1 = \perp$

When $br1$ maps to $R1$, we can have the following possible mappings for $br2$:

$br2 = R2$
$br2 = \perp$

When both model and instance have many individuals of the same types there are many possible role assignments.

For the role assignment: $Apt\_13 = apt1, br1 = R1, bed1 = b1, br2 = \perp$, the semantic network shown in Figure 2 defines a Bayesian network as shown in Figure 3.

The Boolean variable $\langle Apt\_13 \sim apt1 \rangle$ denotes whether model Apt_13 matches with instance $apt1$. The Boolean variable $\langle br1 \sim R1 \rangle$ represents whether individual $br1$ of the model matches with the individual $R1$ of the instance. The Boolean variable $\langle br2, \perp \rangle$ represents whether individual $br2$ of the model does not map to any individual of instance.

The conditional probabilities of the Bayesian network shown in Figure 3 are constructed using the supermodel and Apt_13's description. Some of these probabilities are shown below:

$$\begin{aligned} P(\langle Apt\_13 \sim apt1 \rangle = true) &= p0 \\ P(\langle Apt\_13, hasSize \rangle = \text{“large”} | \langle Apt\_13 \sim apt1 \rangle = true) &= p1 \end{aligned}$$

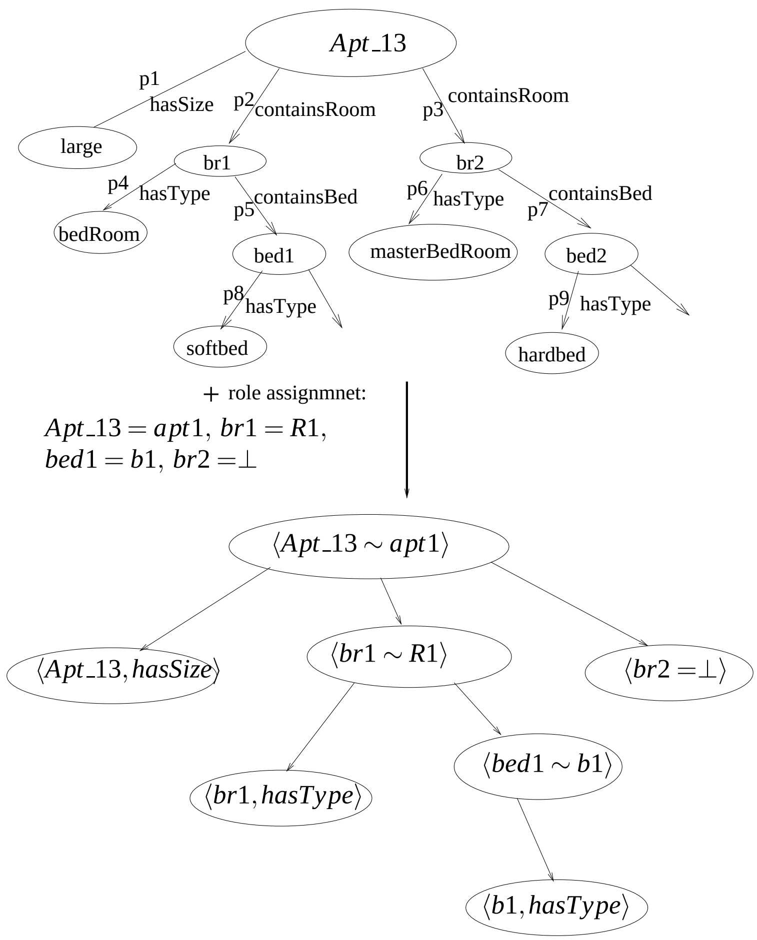

Figure 3: A Bayesian network defined by the semantic network shown in Figure 2 for the role assignment: $Apt\_13 = apt1$, $br1 = R1$, $bed1 = b1$, $br2 = \perp$.

When $\langle Apt\_13 \sim apt1 \rangle = false$, $P(\langle Apt\_13, hasSize \rangle =$ "large"$| \langle Apt\_13 \sim apt1 \rangle = false)$ is the prior probability that Apt_13 is of size "large".

After constructing a Bayesian network, given a role assignment, from the semantic network, we want to compute the posterior probability of $M \sim i$, i.e, $P(M \sim i | observation)$. The observation is the instance i's description, which can be at different level of abstraction than M's description. In the Bayesian network shown in Figure 3, the model individual $br1$ maps to instance individual $R1$. The model specifies $\langle br1, type, bedroom, p4 \rangle$ and instance specifies $\langle R1, type, masterbedroom, present \rangle$, which are at different level of abstraction. To insert the evidence in the constructed Bayesian network, we need to take this difference into consideration. In particular, we need to map the instance description to the evidence for the constructed Bayesian network.

7.2 Mapping an instance description to evidence

An instance description is a set of quadruples of the forms $\langle i_k, prop, V, present \rangle$, and $\langle i_k, prop, V, absent \rangle$. We map the instance description to the evidence for the constructed Bayesian network as follows:

- the quadruple $\langle i_k, prop, V, present \rangle$, if $prop$ is non-functional, provides observation: $\langle i_k, prop, V \rangle = true$ for the constructed Bayesian network.
- the quadruple $\langle i_k, prop, V, absent \rangle$, if $prop$ is non-functional, provides observation: $\langle i_k, prop, V \rangle = false$.
- the quadruple $\langle i_k, prop, v, present \rangle$, if $prop$ is functional and datatype, provides observation: $\langle i_k, prop \rangle = v$.
- the quadruple $\langle i_k, prop, v, present \rangle$, if $prop$ is functional objecttype or $prop$ is $hasType$, provides observation: $\langle i_k, prop \rangle \in v$. We have two cases:
  - If the observation $\langle i_k, prop \rangle \in v$ implies "true" or "false" for the node $\langle M_p, prop \rangle$, such that $M_p = i_k$ exists in the role assignmnet, we do the normal (usual) conditioning in the Bayesian network.
  - If the observation $\langle i_k, prop \rangle \in v$ does not imply "true" or "false" for the node $\langle M_p, prop \rangle$, such that $M_p = i_k$ exists in the role assignment (i.e, v is the superclass of value V of $\langle M_p, prop \rangle$), we provide soft evidence for the node $\langle M_p, prop \rangle$ in the constructed Bayesian network.

    For the soft conditioning, we create an observed child $\langle M_k, prop \rangle$ of $\langle M_p, prop \rangle$, M_k is the same type as M_p. We observed $\langle M_k, prop \rangle \in v$. The conditional probability of $\langle M_k, prop \rangle \in v$ is:

$$
\begin{aligned}
P(\langle M_k, prop \rangle \in v | \langle M_p, prop \rangle \in V) &= 1.0 \\
P(\langle M_k, prop \rangle \in v | \langle M_p, prop \rangle \notin V) &= P(v | \neg V)
\end{aligned}
$$

    Using Bayes rule, probability $P(v | \neg V)$can be computed as follows:

$$
P(v | \neg V) = \frac{P(v) - P(V)}{1 - P(V)}
$$

    where $P(v)$ and $P(V)$ are the probabilities of classes v and V respectively. We can compute $P(v)$ and $P(V)$ using the probabilities associated with the abstraction hierarchies[6] as discussed in Section 5.1.

7.3 Inference

We can compute the posterior probability of match, $P(M \sim i | observation)$, from the constructed Bayesian network using any standard inference algorithms, e.g., VE [Zhang and Poole, 1994]. The posterior probability of match depends on the role assignments of the individuals. We maximize it over all possible role assignments.

8 Conclusion

In this paper, we have proposed a framework for decision making in rich domains, where we can describe the observations (or instances) in the world at multiple levels of abstraction and detail and have probabilistic models at different levels of abstraction and detail, and be able to use them to make decisions. We can build knowledge-based decision tools in various domains such as mineral exploration and hazard mappings, where we need to have probabilistic reasoning and rich ontologies.

Acknowledgments

Rita Sharma was partly supported by a MITACS Postdoctoral fellowship.

References

Costa, P., Laskey, K. and Laskey, K. [2005]. Pr-owl: A bayesian ontology language for the semantic web, *Proceedings of the ISWC Workshop on Uncertainty Reasoning for the Semantic Web*.

Ding, Z. and Peng, Y. [2004]. A probabilistic extension to ontology language owl, *Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04)*.

[6] These are provided by the supermodel.

Hart, P. [1975]. Progress on a computer-based consultant, *Proceedings of International Joint Conference on Artificial Intelligence*, pp. 831–841.

McGuinness, D. and van Harmelen, F. [2004]. Owl web ontology language overview, *W3C Recommendation 10 February 2004, W3C* .

Pool, M., Fung, F., Cannon, S. and Aikin, J. [2005]. Is it worth a hoot? qualms about owl for uncertainty reasoning, *Proceedings of the ISWC Workshop on Uncertainty Reasoning for the Semantic Web*.

Poole, D. and Smyth, C. [2005]. Type uncertainty in ontologically-grounded qualitative probabilistic matching, *Eighth European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU-2005)*, pp. 763–774.

Smyth, C. and Poole, D. [2004]. Qualitative probabilistic matching with hierarchical descriptions, *Proceedings of Ninth International Conference on the Principles of Knowledge Representation and Reasoning (KR-2004)*.

Zhang, N. and Poole, D. [1994]. A simple approach to Bayesian network computation, *Proc. of the 10th Candian Conference on Artificial Intelligence*, pp. 171–178.