A Bayesian Methodology Towards Automatic Ontology Mapping*

Zhongli Ding, Yun Peng, Rong Pan, and Yang Yu

University of Maryland Baltimore County
Department of Computer Science and Electrical Engineering
1000 Hilltop Circle, Baltimore, MD 21250
{zding1, ypeng, panrong1, yangyu1}@cs.umbc.edu

Abstract

This paper presents our ongoing effort on developing a principled methodology for automatic ontology mapping based on *BayesOWL*, a probabilistic framework we developed for modelling uncertainty in semantic web. The proposed method includes four components: 1) learning probabilities (priors about concepts, conditionals between subconcepts and superconcepts, and raw semantic similarities between concepts in two different ontologies) using Naïve Bayes text classification technique, by explicitly associating a concept with a group of sample documents retrieved and selected automatically from World Wide Web (WWW); 2) representing in OWL the learned probability information concerning the entities and relations in given ontologies; 3) using the *BayesOWL* framework to automatically translate given ontologies into the Bayesian network (BN) structures and to construct the conditional probability tables (CPTs) of a BN from those learned priors or conditionals, with reasoning services within a single ontology supported by Bayesian inference; and 4) taking a set of learned initial raw similarities as input and finding new mappings between concepts from two different ontologies as an application of our formalized BN mapping theory that is based on evidential reasoning across two BNs.

Overview

Semantic heterogeneity between two different applications or agents comes from their use of conflicted or mismatched terms about concepts. Same term or concept name might have different meanings in different agents, different terms from different agents might have the same meaning, one term from an agent might matches to several or might not matches to any terms of the other agent exactly, or two terms with the same or similar meaning are structured differently in different agents (e.g., different paths from their respective root concepts). With the development of the semantic web[1], ontologies have become widely used to represent the conceptualization of a domain, i.e., concepts, properties about concepts, relations between concepts, and instances about concepts. In ontology-based semantic integration, two agents in communication need to find a way to share the semantics of the terms in their ontologies in order to fully understand each other. This can be done in several possible directions depends on the needs of particular applications: 1) one may force both agents to use a single centralized global ontology; 2) one may merge the source ontologies into one unified ontology before agent interactions; 3) one may search for a set of mappings (or matches) between two ontologies; 4) for a multi-agent system one may resolve semantic differences in runtime when they arise during agent interaction; and 5) one may translate one of the ontologies into a target ontology with the help of an intermediate shared ontology. In this context, we are particularly interested in ontology mapping. (Noy 2004) provides a brief survey about existing ontology-based approaches, which are either based on syntactic and semantic heuristics, machine learning text classification techniques by attaching a set of documents to each concept to represent its meaning, or linguistics (spelling, lexicon relations, lexical ontologies, etc.) and natural language processing techniques.

Ontology languages in the semantic web, such as OWL[2] and RDF(S)[3], are based on crisp logic and thus can not handle incomplete or partial knowledge about an application domain. However, uncertainty exists in almost every aspects of ontology engineering. For example, in domain modelling, besides knowing that "*A* is a subclass of *B*", one may also know and wishes to express that "*A* is a small subclass of *B*"; or, in the case that *A* and *B* are not logically related, one may still wishes to express that "*A* and *B* are largely overlapped with each other". In ontology reasoning, one may want to know not only if *A* is a subsumer of *B*, but also how close of *A* is to *B*; or, one may want to know the degree of similarity even if *A* and *B* are not subsumed by each other. Moreover, a description (of a class or object) one wishes to input to an ontology reasoner may be noisy and uncertain. Uncertainty becomes more prevalent in concept mapping between two ontologies where it is often the case that a concept defined in one ontology can only find partial matches to one or more concepts in another ontology.

* This work was supported in part by DARPA contract F30602-97-1-0215 and NSF award IIS-0326460.

1 http://www.w3.org/2001/sw/

2 http://www.w3.org/2001/sw/WebOnt/

3 http://www.w3.org/RDF/

Narrowly speaking, a mapping can be defined as a correspondence between concept *A* in Ontology 1 and concept *B* in Ontology 2 which has similar or same semantics as *A*. Most existing ontology-based semantic integration approaches provide exact mappings in a semi-automatic way with manual validation, without taking the degree of uncertainty into consideration. In tackling this problem, (Mitra, Noy and Jaiswal 2004) improves existing mapping results using BNs (Pearl 1988) by a set of meta-rules that capture the structural influence and the semantics of ontology relations.

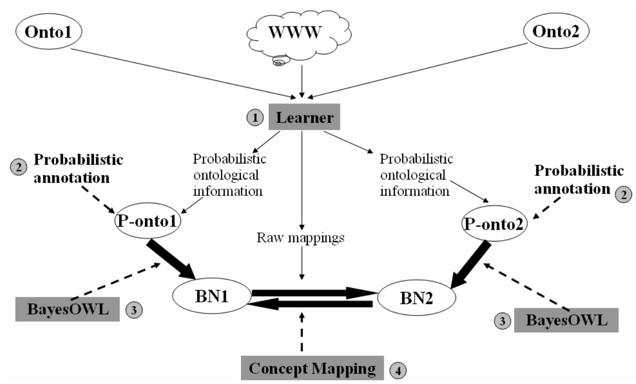


Figure 1. The System Framework

Different from their contributions, we propose a new methodology in supporting uncertainty modelling and reasoning in a single ontology, as well as ontology mapping using Bayesian networks. As can be seen from Figure 1 above, the system includes four components: 1) a learner to obtain probabilistic ontological information and raw mappings using data obtained from web; 2) a representation mechanism for the learned uncertain information concerning the entities and relations in given ontologies; 3) a *BayesOWL* (Ding, Peng, and Pan 2004; Ding and Peng 2004) module to translate given ontologies (together with the learned uncertain information) into BNs; and 4) a concept mapping module which takes a set of learned raw similarities as input and finds mappings between concepts from two different ontologies based on evidential reasoning across two BNs. The ideas about these four components, as well as their related works, are presented in the next four sections respectively. The paper ends with a discussion and suggestions for future research.

Learning Probabilities from Web Data

In this work, we use prior probability distributions *P(C)* to capture the uncertainty about concepts (i.e., how an arbitrary individual belongs to class *C*), conditional probability distributions *P(C|D)* for relations between *C* and *D* in the same ontology (e.g., how likely an arbitrary individual in class *D* is also in *D*'s subclass *C*), and joint probability distributions *P(C,D)* for semantic similarity between concepts *C* and *D* from different ontologies. In many cases these kinds of probabilistic information are not available and are difficult to obtain from domain experts. Our solution is to learn these probabilities using Naïve Bayes text classification technique (Craven et al. 2000; McCallum and Nigam 1998) by associating a concept with a group of sample documents called *exemplars*. The idea is inspired by those machine learning based semantic integration approaches such as (Doan et al. 2002; Lacher and Groh 2001; Prasad, Peng and Finin 2002) where the meaning of a concept is implicitly represented by a set of exemplars that are relevant to it.

Learning the probabilities we need from these exemplars is straightforward. First, we build a model containing statistical information about each concept's exemplars in Ontology 1 using a text classifier such as Rainbow[1], and then classify each concept in Ontology 2 by their respective exemplars using the model of Ontology 1 to obtain a set of probabilistic scores showing the similarity between concepts. Ontology 1's exemplars can be classified in the same way by model built using Ontology 2's exemplars. This cross-classification (Figure 2) process helps find a set of raw mappings between Ontology 1 and Ontology 2 by setting some threshold values. Similarly, we can obtain prior or conditional probabilities related to concepts in a single ontology through self-classification with the model for that ontology.

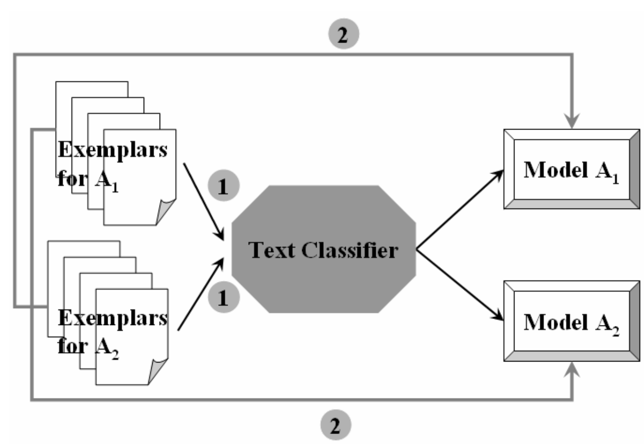


Figure 2. Cross-classification using Rainbow

The quality of these text classification based mapping algorithms is highly dependent on the quality of the exemplars (how relevant they are to the concept and how comprehensive they are in capturing all important aspects of the concept), and it would be a very time-consuming task for knowledge workers to choose high quality exemplars manually. The need to find sufficient relevant exemplars for

[1] http://www-2.cs.cmu.edu/~mccallum/bow/rainbow

a large quantity of concepts manually greatly reduces the attractiveness and applicability of these machine learning based approaches.

Our approach is to use search engines such as Google[1] to retrieve exemplars for each concept node automatically from WWW, the richest information resource available nowadays. The goal is to search for documents in which the concept is used in its intended semantics. The rationale is that the meaning of a concept can be described or defined in the way it is used.

To find out what documents are relevant to a term, one can use words of the term as keywords to query the search engine. However, a word may have multiple meanings (word senses) and a query using only words of the term in attention may return irrelevant documents based on a different meaning of that word. For example, in an ontology for "food", a concept named "apple" is a subconcept of "fruit". If one only uses "apple" as the keyword for query, documents showing how to make an apple pie and documents showing how to use an iPod may both be returned. Apparently, the documents using "apple" for its meaning in computer field is irrelevant to "apple" as a fruit. Fortunately, since we are dealing with concepts in well defined ontologies, the semantics of a term is to a great extent specified by the other terms used in defining this concept in the ontology, names, the properties of that concept class, its super- and sub-concept classes. For example, if a given ontology is a concept taxonomy, the search query can be formed with all the terms on the path from root to the node in the taxonomy. By this method, the number of irrelevant documents returned is greatly reduced. In the "apple" example, the query would then become "food fruit apple" instead of "apple" itself. Documents about iPod and Apple computers will not be returned.

Search results returned by search engines are html files. There are some choices on how to use them. The simplest one is to use the entire html file as one exemplar. A second option is to use each paragraph where a keyword in the query shows up. A third option is to collect sentences containing a keyword in the html file and use this collection as an exemplar. We are currently experimenting these options, and the preliminary results suggest the second approach is the most suitable one.

Representing Probabilities in OWL

Information about the uncertainty of the classes and relations in an ontology can often be represented as probability distributions (e.g., $P(C)$ and $P(C|D)$ mentioned earlier), which we refer to as *probabilistic constraints* on the ontology. These probabilities can be either provided by domain experts or learned from web data as described in the previous section.

[1] http://www.google.com

Although not necessary, it is beneficial to represent the probabilistic constraints as OWL statements. We have developed such a representation. At the present time, we only provide encoding of two types of probabilities: priors and pair-wise conditionals. This is because they correspond naturally to classes and relations (RDF triples) in an ontology, and are most likely to be available to ontology designers. The representation can be easily extended to constraints of other more general forms if needed.

The model-theoretic semantics of OWL treats the domain as a non-empty collection of individuals. If class A represents a concept, we treat it as a random binary variable of two states a and $\overline{a}$, and interpret $P(A=a)$ as the prior probability or one's belief that an arbitrary individual belongs to class A, and $P(a|b)$ as the conditional probability that an individual of class B also belongs to class A. Similarly, we can interpret $P(\overline{a})$, $P(\overline{a}|b)$, $P(a|\overline{b})$, and $P(\overline{a}|\overline{b})$ with the negation interpreted as "not belonging to".

We treat a probability as a kind of resource, and define two OWL classes: "PriorProb" and "CondProb". A prior probability of a variable is defined as an instance of class "PriorProb", which has two mandatory properties: "hasVarible" (only one) and "hasProbValue" (only one). A conditional probability of a variable is defined as an instance of class "CondProb" with three mandatory properties: "hasCondition" (at least has one), "hasVariable" (only one), and "hasProbValue" (only one).

The range of "hasCondition" and "hasVariable" is a defined class named "Variable" with two mandatory properties: "hasClass" and "hasState". "hasClass" points to the concept class this probability is about and "hasState" gives the "True" (belong to) or "False" (not belong to) state of this probability.

For example, $P(c)=0.3$, the prior probability that an arbitrary individual belongs to class C, can be expressed as

```
<Variable rdf:ID="c">
  <hasClass>C</hasClass>
  <hasState>True</hasState>
</Variable>
<PriorProb rdf:ID="P(c)">
  <hasVariable>c</hasVariable>
  <hasProbValue>0.3</hasProbValue>
</PriorProb>
```

and conditional probability $P(c\,|\,p1,p2)=0.8$ can be encoded as

```
<CondProb rdf:ID="P(c|p1, p2)">
  <hasCondition>p1</hasCondition>
  <hasCondition>p2</hasCondition>
  <hasVariable>c</hasVariable>
  <hasProbValue>0.8</hasProbValue>
</CondProb>
```

with variables c, p1, and p2 properly defined.

Similar to our work, (Fukushige 2004) proposes a vocabulary for representing probabilistic relationships in a RDF graph. Three kinds of probability information can be encoded in his framework: probabilistic relations (prior), probabilistic observation (data), and probabilistic belief (posterior). And any of them can be represented using probabilistic statements which are either conditional or unconditional.

The *BayesOWL* Framework

BayesOWL (Ding, Peng and Pan 2004; Ding and Peng 2004) is a framework which augments and supplements OWL for representing and reasoning with uncertainty, based on Bayesian networks (BN). This framework provides a set of rules and procedures for direct translation of an OWL ontology into a BN structure and a method that incorporate encoded probability information when constructing the conditional probability tables (CPTs) of the BN. The translated BN, which preserves the semantics of the original ontology and is consistent with the probability information, can support ontology reasoning, both within and across ontologies as Bayesian inferences. Below we give a brief summary.

Structural Translation

A set of translation rules is developed to convert an OWL ontology (about TBox only at the present time) into a directed acyclic graph (DAG) of BN. The general principle underlying these rules is that all classes (specified as "subjects" and "objects" in RDF triples of the OWL file) are translated into nodes in BN, and an arc is drawn between two nodes in BN if the corresponding two classes are related by a "predicate" in the OWL file, with the direction from the superclass to the subclass. Control nodes are created during the translation to facilitate modelling relations among class nodes that are specified by OWL *logical* operators, and there is a converging connection from each concept nodes involved in this logical relation to its specific control node. There are five types of control nodes in total, which correspond to the five types of logical relations: "and" (owl:intersectionOf), "or" (owl:unionOf), "not" (owl:complementOf), "disjoint" (owl:disjointWith), and "same as" (owl:equivalentClass).

Constructing CPTs

The nodes in the DAG obtained from the structural translation step can be divided into two disjoint groups: X_R, nodes representing concepts in ontology, and X_C, control nodes for bridging logical relations. The CPT for a control node in X_C can be determined by the logical relation it represents so that when its state is "True", the corresponding logical relation holds among its parent nodes. When all the control nodes' states are set to "True" (denote this situation as *CT*), all the logical relations defined in the original ontology are held in the translated BN. The remaining issue is then to construct the CPTs for each node in X_R so that $P(X_R|CT)$, the joint distribution of all regular nodes in the subspace of *CT*, is consistent with all the given probabilistic constraints (which can be learned from web data as described earlier).

This is difficult for two reasons. First, the constraints are usually not given in the form of CPT. For example, CPT for variable C with two parents A and B is in the form of $P(C|A,B)$ but a constraint may be given as $Q(C|A)$ or even $Q(C)$. Secondly, CPTs are given in the general space of $X = X_R \cup X_C$, but constraints are for the subspace of *CT* (the dependencies changes when going from the general space to the subspace of *CT*). For example, with the constraint $Q(C|A)$, $P(C|A,B)$, the CPT for C, should be constructed in such a way that $P(C|A,CT) = Q(C|A)$. To overcome these difficulties, we developed an algorithm named D-IPFP (Ding, Peng, and Pan 2004) to approximate these CPTs for X_R based on the "iterative proportional fitting procedure" (IPFP), a well-known mathematical procedure that modifies a given distribution to meet a set of probabilistic constraints while minimizing *I-divergence* to the original distribution (Deming and Stephan 1940; Csiszar 1975; Bock 1989; Vomlel 1999; Cramer 2000).

Figure 3 below is a BN translated from a simple ontology. In this ontology, "Animal" is a primitive concept class; "Male", "Female", "Human" are **subclasses** of "Animal"; "Male" and "Female" are **disjoint** with each other; "Man" is the **intersection** of "Male" and "Human"; "Woman" is the **intersection** of "Female" and "Human"; "Human" is the **union** of "Man" and "Woman".

The following probability constraints are attached to X_R = {Animal, Male, Female, Human, Man, Woman}:

P(Animal) = 0.5; P(Male|Animal) = 0.5;
P(Female|Animal) = 0.48; P(Human|Animal) = 0.1;
P(Man|Human) = 0.49; P(Woman|Human) = 0.51.

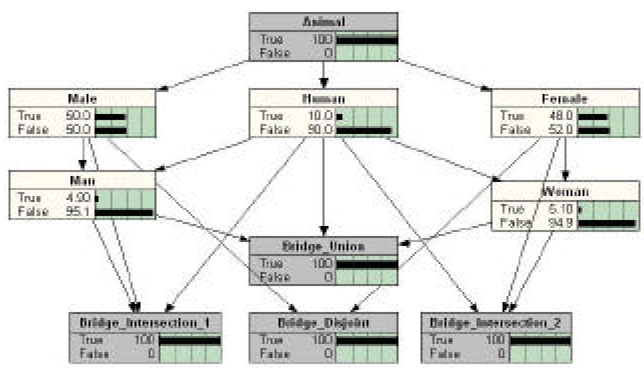


Figure 3. A Translation Example

Reasoning within Single Ontology

The *BayesOWL* framework can support common ontology reasoning tasks as probabilistic inferencesg in the translated BN, for example, given a concept description e, it can answer queries about concept satisfiability (whether $P(e|CT) = 0$), about concept overlapping (how close e is to a concept *C* as *P(e|C,CT)*), and about concept subsumption (find the concept which is most similar to *e*) by defining some similarity measures such as Jaccard Coefficient (Rijsbergen 1979).

Prototype Implementation

A prototype system named *OWL2BN* (Figure 4) is currently under active construction. It takes a valid OWL ontology and some consistent probabilistic constraints as input and outputs a translated BN, with reasoning services provided based on BN inference methods.

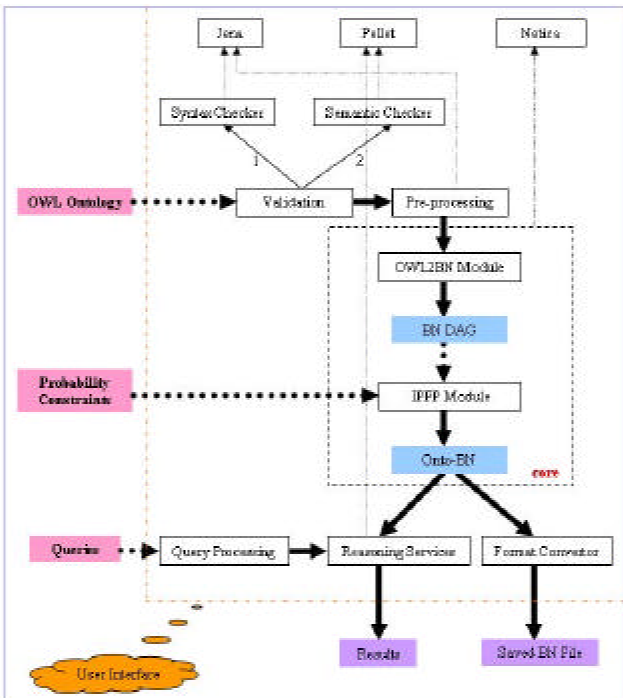


Figure 4. *OWL2BN*: Implementation of *BayesOWL*

Comparison to Related Works

Many of the suggested approaches to quantify the degree of overlap or inclusion between two concepts are based on ad hoc heuristics, others combine heuristics with different formalisms such as fuzzy logic, rough set theory, and Bayesian probability (see (Stuckenschmidt and Visser 2000) for a brief survey). Among them, works that integrate probabilities with description logic (DL) based systems are most relevant to *BayesOWL*. This includes probabilistic extensions to *ALC* based on probabilistic logics (Heinsohn 1994, Jaeger 1994); P-*SHOQ*(D) (Giugno and Lukasiewicz 2002), a probabilistic extension of *SHOQ*(D) based on the notion of probabilistic lexicographic entailment; and several works on extending DL with Bayesian networks (P-CLASSIC (Koller et al. 1997) that extends CLASSIC, PTDL (Yelland 1999) that extends TDL (Tiny Description Logic with only " Conjunction" and " Role Quantification" operators), and the work of Holi and Hyvö nen (2004) which uses BN to model the degree of subsumption for ontologies encoded in RDF(S)).

The works closest to ours in this field are P-CLASSIC and PTDL. In contrast to these works, one of *BayesOWL*'s major contribution is its D-IPFP mechanism to construct CPTs from given piece-wised probability constraints. Moreover, in *BayesOWL*, by using control nodes, the "rdfs:subclassOf" relations (or the subsumption hierarchy) are separated from other logical relations, so the in-arcs to a regular concept node *C* will only come from its parent superclass nodes, which makes *C*'s CPT smaller and easier to construct than P-CLASSIC or PTDL, especially in a domain with rich logical relations.

Also, *BayesOWL* is not to extend or incorporate into OWL or any other ontology language or logics with probability theory, but to translate a given ontology to a BN in a systematic and practical way, and then treats ontological reasoning as probabilistic inferences in the translated BNs. Several benefits can be seen with this approach. It is non-intrusive in the sense that neither OWL nor ontologies defined in OWL need to be modified. Also, it is flexible, one can translate either the entire ontology or part of it into BN depending on the needs. Moreover, it does not require availability of complete conditional probability distributions, pieces of probability information can be incorporated into the translated BN in a consistent fashion. With these and other features, the cost of our approach is low and the burden to the user is minimal. We also want to emphasis that *BayesOWL* can be easily extended to handle other ontology representation formalisms (syntax is not important, semantic matters), if not using OWL.

Concept Mapping between Ontologies as an Application of BN Mapping

It is often the case when attempting to map concept *A* defined in Ontology 1 to Ontology 2 there is no concept in Ontology 2 which is semantically identical to *A*. Instead, *A* is similar to several concepts in Ontology 2 with different degree of similarity. A solution to this so-called one-to-many problem, as suggested by (Prasad, Peng, and Finin 2002) and (Doan et al. 2003), is to map *A* to the target concept *B* which is most similar to *A* by some measure. This simple approach would not work well because 1) the degree of similarity between *A* and *B* is not reflected in *B* and thus will not be considered in reasoning after the mapping; 2) it cannot handle the situation where *A* itself is uncertain; and 3) potential information loss because other similar concepts are ignored in the mapping.

With *BayesOWL*, concept mapping can be processed as some form of probabilistic evidential reasoning between the BN1 and BN2, translated from the Ontologies 1 and 2. This may allow us to address some of the aforementioned difficulties by utilizing BN techniques for integrating probabilistic knowledge and information from various sources. This section will first present a framework of variable mapping between BNs, before illustrating how ontology mapping can be conducted using this framework.

BN Mapping Framework

In applications on large, complex domains, often separate BNs describing related subdomains or different aspects of the same domain are created, but it is difficult to combine them for problem solving — even if the interdependency relations are available. This issue has been investigated in several works, including most notably Multiply Sectioned Bayesian Network (MSBN) by Xiang (2002) and Agent Encapsulated Bayesian Network (AEBN) by Valtorta *et al.* (2002). However, their results are still restricted in scalability, consistency and expressiveness. MSBN's pair-wise variable linkages are between identical variables with the same distributions, and, to ensure consistency, only one side of the linkage has a complete CPT for that variable. AEBN also requires a connection between identical variables, but allows these variables to have different distributions. Here, identical variables are the same variables reside in different BNs.

What we need in supporting mapping concepts is a framework that allows two BNs (translated from two ontologies) to exchange beliefs via variables that are similar but not identical. We illustrate our ideas by first describing how mapping shall be done for a pair of similar concepts (A from ontology 1 to B in ontology 2), and then discussing how such pair-wise mappings can be generalized to network to network mapping. We assume the similarity information between A and B is captured by the joint distribution $P(A, B)$.

Now we are dealing with three probability spaces: S_A and S_B for BN1 and BN2, and S_{AB} for $P(A, B)$. The mapping from A to B amounts to determine the distribution of B in S_B, given the distribution $P(A)$ in S_A under the constraint $P(A, B)$ in S_{AB}.

To propagate probabilistic influence across these spaces, we can apply Jeffrey's rule and treat the probability from the source space as soft evidence to the target space (Pearl, 1990, Valtorta et al., 2002). The rule is given in (1), where Q denotes probabilities associated with soft evidence

(1) $Q(Y) = \sum_i P(Y \mid X_i) Q(X_i)$.

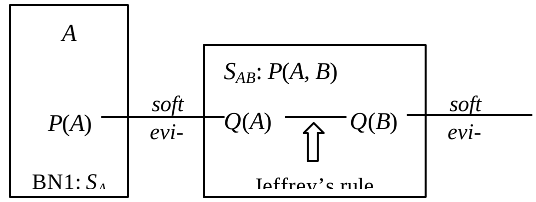
As depicted in Figure 5, mapping A to B is accomplished by applying Jeffrey's rule twice, first from S_A to S_{AB}, then S_{AB} to S_B. Since A in S_A is identical to A in S_{AB}, $P(A)$ in S_A becomes soft evidence $Q(A)$ to S_{AB} and by (1), the distribution of B in S_{AB} is updated to

(2) $Q(B) = \sum_i P(B \mid A_i) Q(A_i)$.

$Q(B)$ is then applied as soft evidence from S_{AB} to node B in S_B, updating beliefs for every variable V in S_B by

(3) $Q(V) = \sum_j P(V \mid B_j) Q(B_j)$

$$= \sum_j P(V \mid B_j) \sum_i P(B_j \mid A_i) P(A_i)$$



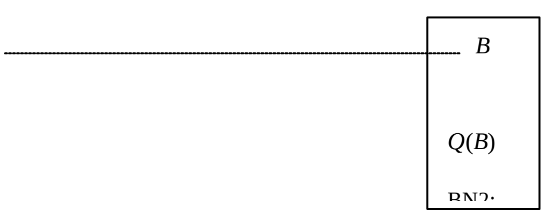


Figure 5. Mapping concept A to B

Back to the example in Figure 3, where the posterior distribution $P(Human \mid \neg Male \cap Animal)$ is (0.102, 0.898). Suppose we have another BN with a variable "Adult" with marginal distribution (0.8, 0.2). Suppose we also know that "Adult" is similar to "Human" with conditional distribution

$$P(Adult \mid Human) = \begin{pmatrix} 0.7 & 0.3 \\ 0 & 1 \end{pmatrix}.$$

Mapping "Human" to "Adult" leads to a change of latter's distribution from (0.8, 0.2) to (0.0714, 0.9286). This change can then be propagated to further update believes of all other variables in the target BN by (3).

Mapping Reduction

A pair-wise linkage as described above provides a channel to propagate belief from A in one BN to influence the belief of B in another BN. When the propagation is completed, (2) must hold between the distributions of A and B. If there are multiple such linkages, (2) must hold simultaneously for all pairs. In theory, any pair of variables between two BNs can be linked, albeit with different degree of similarities. Therefore we may potentially have $n_1 \cdot n_2$ linkages (n_1 and n_2 are the number of variables in BN1 and BN2, respectively). Although we can update the distribution of BN2 to satisfy all linkages by IPFP using (2) as constraints, it would be a computational formidable task.

Fortunately, satisfying a given probabilistic relation between $P(A, B)$ does not require the utilization, or even the existence, of a linkage from A to B. Several probabilistic relations may be satisfied by one linkage. As shown in Figure 6, we have variables A and B in BN_1, C and D in BN_2, and probability relations between every pair as below:

$$P(C, A) = \begin{pmatrix} 0.3 & 0 \\ 0.1 & 0.6 \end{pmatrix}, P(D, A) = \begin{pmatrix} 0.33 & 0.18 \\ 0.07 & 0.42 \end{pmatrix},$$

$$P(D, B) = \begin{pmatrix} 0.348 & 0.162 \\ 0.112 & 0.378 \end{pmatrix}, P(C, B) = \begin{pmatrix} 0.3 & 0 \\ 0.16 & 0.54 \end{pmatrix}.$$

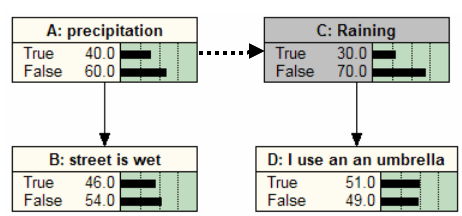


Figure 6. Mapping Reduction Example

However, we do not need to set up linkages for all these relations. As Figure 6 depicts, when we have a linkage from *A* to *C*, all these relations are satisfied (the other three linkages are thus redundant). This is because not only beliefs on *C*, but also beliefs on D are properly updated by the mapping *A* to *C*.

Several experiments with large BNs have shown that only very small portions of all $n_1 \cdot n_2$ linkages are needed in satisfying all probability constraints. This, we suspect, is due to the fact that some of these constraints can be derived from others based on the probabilistic interdependencies among variables in the two BN. We are currently actively working on developing a set of rules that examine the BN structures and CPTs so that redundant linkages can be identified and removed.

Discussion and Future Work

This paper describes our ongoing research on developing a probabilistic framework for automatic ontology mapping. In this framework, ontologies (or parts of them) are first translated into Bayesian networks, then the concept mapping is realized as evidential reasoning between the two BNs by Jeffrey's rule. The probabilities needed in both translation and mapping can be obtained by using text classification programs, supported by associating to individual relevant text exemplars retrieved from the web.

We are currently actively working on each of these components. In searching for relevant exemplar, we are attempting to develop a measure of relevancy so that less relevant documents can be removed. We are expanding the ontology to BN translation from taxonomies to include properties, and develop algorithms to support common ontology-related reasoning tasks. As for a general BN mapping framework, our current focus is on linkage reduction. We are also working on the semantics of BN mapping and examine its scalability and applicability.

Future work also includes developing methods in handling inconsistent probability constraints. The study of IPFP also motivated us to develop a new algorithm named E-IPFP (Peng and Ding 2005). This algorithm is more general than the D-IPFP algorithm we used for constructing CPTs in ontology to BN translation in that it can accommodate any types of probability constraint, not only priors and pair-wise conditionals. We are working on a new algorithm that combines both E-IPFP and D-IPFP for a computationally efficient construction of CPTs for general BN.

Acknowledgement

This work was supported in part by DARPA contract F30602-97-1-0215 and NSF award IIS-0326460.

References

Bock, H. H. 1989. A Conditional Iterative Proportional Fitting (CIPF) Algorithm with Applications in the Statistical Analysis of Discrete Spatial Data. *Bull. ISI, Contributed Papers of 47th Session in Paris*, 1: 141-142.

Cramer, E. 2000. Probability Measures with Given Marginals and Conditionals: *I*-projections and Conditional Iterative Proportional Fitting. *Statistics and Decisions*, 18: 311-329.

Craven, M.; DiPasquo, D.; Freitag, D.; McCallum, A.; Mitchell, T.; Nigam, K.; and Slattery, S. 2000. Learning to Construct Knowledge Bases from the World Wide Web. *Artificial Intelligence*, 118(1-2): 69-114.

Csiszar, I. February 1975. *I*-divergence Geometry of Probability Distributions and Minimization Problems. *The Annuals of Probability*, 3(1): 146-158.

Deming, W. E.; and Stephan, F. F. 1940. On a Least Square Adjustment of a Sampled Frequency Table when the Expected Marginal Totals are Known. *Ann. Math. Statist.* 11: 427-444.

Ding, Z.; Peng, Y.; and Pan, R. November 2004. A Bayesian Approach to Uncertainty Modelling in OWL Ontology. In *Proceedings of 2004 International Conference on Advances in Intelligent Systems - Theory and Applications (AISTA2004)*. Luxembourg-Kirchberg, Luxembourg.

Ding, Z.; and Peng, Y. January 2004. A Probabilistic Extension to Ontology Language OWL. In *Proceedings of the 37th Hawaii International Conference on System Sciences (HICSS-37)*. Big Island, Hawaii.

Doan, A. H.; Madhavan, J.; Domingos, P.; and Halevy, A. May 2002. Learning to Map between Ontologies on the Semantic Web. In *WWW 2002*. Honolulu, Hawaii, USA.

Fukushige, Y. October 2004. Representing Probabilistic Knowledge in the Semantic Web. Position paper for *the W3C Workshop on Semantic Web for Life Sciences*. Cambridge, MA, USA.

Giugno, R.; and Lukasiewicz, T. April 2002. P-*SHOQ*(D): A Probabilistic Extension of *SHOQ*(D) for Probabilistic Ontologies in the Semantic Web. INFSYS Research Report 1843-02-06, Wien, Austria.

Heinsohn, J. 1994. Probabilistic Description Logics. In *Proceedings of UAI-94*, 311-318.

Holi, M.; and Hyvönen, E. 2004. Probabilistic Information Retrieval Based on Conceptual Overlap in Semantic Web Ontologies. In *Proceedings of the 11th Finnish AI Conference, Web Intelligence*, vol. 2, Finnish AI Society, Finland.

Jaeger, M. 1994. Probabilistic Reasoning in Terminological Logics. In *Proceedings of KR-94*, 305-316.

Koller, D.; Levy, A.; and Pfeffer, A. 1997. P-CLASSIC: A Tractable Probabilistic Description Logic. In *Proceedings of AAAI-97*, 390-397.

Lacher, M.; and Groh, G. May 2001. Facilitating the Exchange of Explicit Knowledge through Ontology Mappings. In *Proceedings of the 14th International FLAIRS Conference*. Key West, FL, USA.

McCallum, A.; and Nigam, K. 1998. A Comparison of Event Models for Naive Bayes Text Classification. *AAAI-98 Workshop on "Learning for Text Categorization"*.

Mitra, P.; Noy, N. F.; and Jaiswal, A. R. 2004. OMEN: A Probabilistic Ontology Mapping Tool. In *Workshop on Meaning Coordination and Negotiation at the Third International Conference on the Semantic Web (ISWC-2004)*. Hisroshima, Japan.

Noy, N. F. 2004. Semantic Integration: A Survey Of Ontology-Based Approaches. *SIGMOD Record*, Special Issue on Semantic Integration, 33 (4).

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

Pearl, J. 1990. Jeffery's Rule, Passage of Experience, and neo-Bayesianism. In H.E. et al. Kyburg, Jr., editor, *Knowledge Representation and Defeasible Reasoning*, 245-265.

Peng, Y.; and Ding, Z. July 2005. *Modifying Bayesian Networks by Probability Constraints*. Submitted to UAI 2005. Edinburgh, Scotland.

Prasad, S.; Peng, Y.; and Finin, T. July 2002. A Tool For Mapping Between Two Ontologies Using Explicit Information. In *AAMAS '02 Workshop on Ontologies and Agent Systems*. Italy.

Stuckenschmidt, H.; and Visser, U. 2000. Semantic Translation based on Approximate Re-classification. In *Proceedings of the Workshop "Semantic Approximation, Granularity and Vagueness", KR'00*.

Valtorta, M.; Kim, Y.; and Vomlel, J. 2002. Soft Evidential Update for Probabilistic Multiagent Systems. *International Journal of Approximate Reasoning*, 29(1): 71-106.

van Rijsbergen, C. J. 1979. *Information Retrieval*. London: Butterworths. Second Edition.

Vomlel, J. December 1999. Methods of Probabilistic Knowledge Integration. PhD Thesis, Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University.

Xiang, Y. 2002. *Probabilistic Reasoning in Multiagent Systems: A Graphical Models Approach*. Cambridge University Press.

Yelland, P. M. August 1999. Market Analysis Using a Combination of Bayesian Networks and Description Logics. Sun Microsystems Technical Report TR-99-78.