A reasoner for generalized Bayesian dl-programs

Livia Predoiu

Institute of Computer Science, University of Mannheim, Germany
livia@informatik.uni-mannheim.de

Content

Motivation

- Knowledge in the Semantic Web is provided on independent peers
- Domains overlap, but no (global) reference ontology
- Mappings need to be created dynamically and automatically.
- Automatically created mappings are uncertain hypotheses (oversimplifying, erroneous)
- Uncertainty can be modelled with probabilities

Motivation cont'd

Integrated reasoning with ontologies and uncertain mappings provides

- insight into the (un)certainty of the reasoning results
- a natural ranking method over the reasoning results

Syntax:

- A generalized dl-program $KB = (L, P)$ consists of a description logic knowledge base L in the DLP fragment and a Datalog program P

Semantics:

- L is translated into its Datalog equivalent L'
- $L' \cup P$ are interpreted as Datalog programs

Syntax

- A generalized Bayesian dl-program is a 4-tuple $KB = (L, P, \mu, Comb)$ where
  - (L, P) is a generalized dl-program
  - $\mu(r, v)$ is a probability function over all truth valuations w of the head atom associated with each rule r in $ground(P)$ and every truth valuation v of the body atoms of r
  - $Comb$ is a combining rule, which defines how rules of $r \in ground(P)$ with same head atom can be combined.

Semantics

- each generalized Bayesian dl-program $KB = (L, P, \mu, Comb)$ encodes the structure of a Bayesian Network *BN*
- Translation from *KB* to *BN*
  - (L, P) is translated into its Datalog equivalent $D = L' \cup P$
  - a ground atom *a* is active iff it belongs to the canonical model of *D*; $r \in ground(D)$ is active iff all its atoms are active
  - every active atom corresponds to a node in *BN*
  - every active rule corresponds to direct influence relations betweent the atoms involved and is translated to arcs in *BN*
  - μ is the conditional probability density for each active rule
  - for at least 2 active rules with same head, the combining rule *Comb* generates a joint conditional distribution from the individual ones of the involved rules.

Using generalized Bayesian dl-programs for reasoning with Ontologies and uncertain mappings

Intuitively

- $L = O_1 \cup O_2$ encodes the ontologies
- P, μ encodes the mappings

Mappings:

- $Q(O_i)$ denotes the matchable elements of the ontology O_i
- Matching: Given two ontologies O and O', determine correspondences between $Q(O)$ and $Q(O')$.
- Correspondences are 5-tuples (id, e, e', r, n) such that
  - id is a unique identifier;
  - $e \in Q(O)$ and $e' \in Q(O')$;
  - $r \in R$ is a semantic relation (here: implication);
  - n is a degree of confidence in the correctness. (here: probabilities)

Ontology 1

(1) $Technical\_Report \sqsubseteq \forall keyword.Topic \sqcap \forall author.Person$;
(2) $Book \sqsubseteq Publication$;
(3) $Article \sqsubseteq Publication$;
(4) $Collection \sqsubseteq Publication$;
(5) $Publication \sqsubseteq \forall keyword.Topic \sqcap \forall author.Person$.

Translation of Ontology 1

(1a) $Topic(y) \leftarrow Technical\_Report(x) \wedge keyword(x,y)$;
(1b) $Person(y) \leftarrow Technical\_Report(x) \wedge author(x,y)$;
(2) $Publication(x) \leftarrow Book(x)$;
(3) $Publication(x) \leftarrow Article(x)$;
(4) $Publication(x) \leftarrow Collection(x)$;
(5a) $Topic(y) \leftarrow Publication(x) \wedge keyword(x,y)$;
(5b) $Person(y) \leftarrow Publication(x) \wedge (x,y)$.

Ontology 2

(1) $Paper \sqsubseteq Publication$;
(2) $Proceedings \sqsubseteq Publication$;
(3a) $\top \sqsubseteq \forall includes.Paper$;
(3b) $\top \sqsubseteq \forall includes^{-1}.Proceedings$;
(4a) $\top \sqsubseteq \forall published\_by.Publisher$;
(4b) $\top \sqsubseteq \forall published\_by^{-1}.Publication$;
(5a) $\top \sqsubseteq \forall about.Subject$;
(5b) $\top \sqsubseteq \forall about^{-1}.Publication$;
(6a) $\top \sqsubseteq \forall author.Person$;
(6b) $\top \sqsubseteq \forall author^{-1}.Publication$.

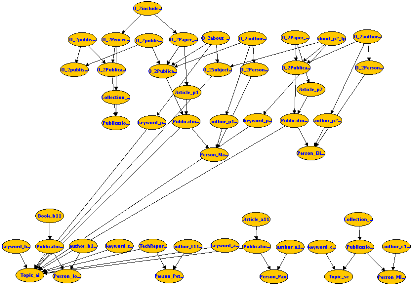
Translation of Ontology 2

(1) $Publication(x) \leftarrow Paper(x)$;
(2) $Publication(x) \leftarrow Proceedings(x)$;
(3a) $Paper(y) \leftarrow includes(x,y)$;
(3b) $Proceedings(x) \leftarrow includes(x,y)$;
(4a) $Publisher(y) \leftarrow published\_by(x,y)$;
(4b) $Publication(x) \leftarrow published\_by(x,y)$;
(5a) $Subject(y) \leftarrow about(x,y)$;
(5b) $Publication(x) \leftarrow about(x,y)$;
(6a) $Person(y) \leftarrow author(x,y)$;
(6b) $Publication(x) \leftarrow author(x,y)$.

Mappings

(1) $O_1' : Publication(x) \overset{(0.9,0.2)}{\longleftarrow} O_2' : Publication(x)$;
(2) $O_1' : Article(x) \overset{(0.7,0.2)}{\longleftarrow} O_2' : Paper(x)$;
(3) $O_1' : Person(x) \overset{(0.9,0.2)}{\longleftarrow} O_2' : Person(x)$;
(4) $O_1' : Collection(x) \overset{(0.7,0.2)}{\longleftarrow} O_2' : Proceedings(x)$;
(5) $O_1' : keyword(x,y) \overset{(0.7,0.2)}{\longleftarrow} O_2' : about(x,y)$;
(6) $O_1' : author(y,x) \overset{(0.7,0.2)}{\longleftarrow} O_2' : author(x,y)$.

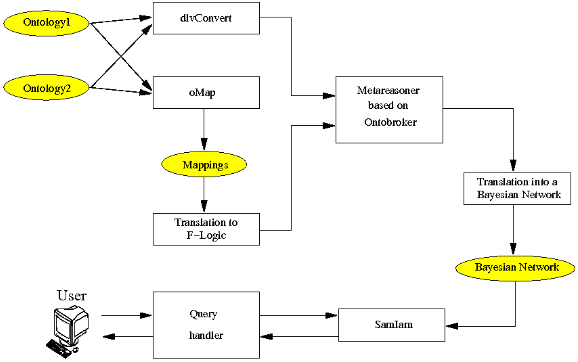
Example Network:



Two types of queries:

- ground queries
- non-ground queries (information retrieval)

Reasoner Architecture



Conclusions and Outlook:

- ongoing implementation of a reasoner for integrated reasoning with deterministic ontologies and probabilistic mappings
- Next steps
  - finish implementation
  - experiments
  - add support for and experiment with more than two ontologies (mapping chains)
  - enhance the efficiency of the reasoning procedure (e.g. by approximation)
    - logical
    - probabilistic
    - Combination of logical & probabilistic
  - add further matchers & trust
  - add mapping conflict resolution