Learning Sentences and Assessments in Probabilistic Description Logics

José Ochoa Luna¹  Kate Revoredo²  Fabio Cozman¹

Decision Making Lab.
Escola Politécnica - Universidade de São Paulo

Departamento de Informática Aplicada
Unirio

URSW 2010
Outline

1. Introduction
2. Probabilistic Description Logic CR\(\mathcal{ALC}\)
3. Learning Description Logics
4. Learning CR\(\mathcal{ALC}\)
5. Preliminary Results
6. Conclusions
Representation of uncertainty in the semantic Web can be favoured by the use of learning techniques
Motivation

- Representation of uncertainty in the semantic Web can be favoured by the use of learning techniques.
- Caveats in syntax and semantics in PDL have prevented them from spreading into several domains.
Motivation

- Representation of uncertainty in the semantic Web can be favoured by the use of learning techniques.
- Caveats in syntax and semantics in PDL have prevented them from spreading into several domains.
- It can be hard to elicit the probability component of a particular set of sentences.
Motivation

- **Representation of uncertainty in the semantic Web can be favoured by the use of learning techniques**
- **Caveats in syntax and semantics in PDL have prevented them from spreading into several domains**
- **It can be hard to elicit the probability component of a particular set of sentences**
- **Focus in CRALC language**
Previous Efforts

Focus on Concept definitions
Using Noisy-OR classifiers
Introduction

Previous Efforts

Focus on Concept definitions
Using Noisy-OR classifiers

Focus on Probabilistic inclusions

\[ C \equiv A \cup B \rightarrow P(C|A \cup B), P(C|A), P(C|B) \]
Combined approach → an algorithm for learning concept definitions and probabilistic inclusions at once
Idea

Combined approach → an algorithm for learning concept definitions and probabilistic inclusions at once

Mostly based on inductive logic programming techniques with a probabilistic twist
Introduction

Idea

Combined approach → an algorithm for learning concept definitions and probabilistic inclusions at once

Mostly based on inductive logic programming techniques with a probabilistic twist

- A search for the best concept description is performed.
Introduction

Idea

Combined approach → an algorithm for learning concept definitions and probabilistic inclusions at once

Mostly based on inductive logic programming techniques with a probabilistic twist

- A search for the best concept description is performed.
- A decision is made as to whether to consider the concept definition found or to insert a probabilistic inclusion based on this concept
**Probabilistic Description Logic CR\textit{ALC}**

\textbf{CR\textit{ALC}} is a probabilistic extension of the DL \textit{ALC}.
Probabilistic Description Logic \textit{cRALC}

- \textit{cRALC} is a probabilistic extension of the DL \textit{ALC}.
- The following constructors are available in \textit{ALC}: \textit{conjunction} (C \sqcap D), \textit{disjunction} C \sqcup D, \textit{negation} (\neg C), \textit{existential} restriction (\exists r.C), and \textit{value restriction} (\forall r.C).
Probabilistic Inclusions and their Semantics

\[ P(A|B) = \alpha \]
Probabilistic Inclusions and their Semantics

- \( P(A|B) = \alpha \)
- \( \forall x \in D : P(A(x)|B(x)) = \alpha \)
Probabilistic Description Logic \textsc{cRA\textit{LC}}

Probabilistic Inclusions and their Semantics

- $P(A|B) = \alpha$
- $\forall x \in \mathcal{D} : P(A(x)|B(x)) = \alpha$
- $P(\text{Professor}(\text{Maria})|\text{Researcher}(\text{Maria})) = 0.4$
Example

\[ P(\text{Animal}) = 0.9, \]
\[ P(\text{Rational}) = 0.6, \]
\[ P(\text{hasChild}) = 0.3, \]
\[ \text{Human} \equiv \text{Animal} \sqcap \text{Rational}, \]
\[ \text{Beast} \equiv \text{Animal} \sqcap \neg \text{Rational}, \]
\[ \text{Parent} \equiv \]
\[ \text{Human} \sqcap \exists \text{hasChild}.\text{Human}, \]
\[ P(\text{Kangaroo}|\text{Beast}) = 0.4, \]
\[ P(\text{Kangaroo}|\neg\text{Beast}) = 0.0, \]
\[ \text{MaternityKangaroo} \equiv \]
\[ \text{Kangaroo} \sqcap \exists \text{hasChild}.\text{Kangaroo} \]
Example

\[ P(\text{Animal}) = 0.9, \]
\[ P(\text{Rational}) = 0.6, \]
\[ P(\text{hasChild}) = 0.3, \]

\[ \text{Human} \equiv \text{Animal} \sqcap \text{Rational}, \]
\[ \text{Beast} \equiv \text{Animal} \sqcap \neg \text{Rational}, \]
\[ \text{Parent} \equiv \]
\[ \text{Human} \sqcap \exists \text{hasChild}.\text{Human}, \]
\[ P(\text{Kangaroo|Beast}) = 0.4, \]
\[ P(\text{Kangaroo|\neg Beast}) = 0.0, \]

\[ \text{MaternityKangaroo} \equiv \]
\[ \text{Kangaroo} \sqcap \exists \text{hasChild}.\text{Kangaroo} \]
Example

\[ P(\text{Animal}) = 0.9, \]
\[ P(\text{Rational}) = 0.6, \]
\[ P(\text{hasChild}) = 0.3, \]

\( \text{Human} \equiv \text{Animal} \sqcap \text{Rational}, \)
\( \text{Beast} \equiv \text{Animal} \sqcap \neg \text{Rational}, \)
\( \text{Parent} \equiv \)
\( \text{Human} \sqcap \exists \text{hasChild. Human}, \)
\[ P(\text{Kangaroo} | \text{Beast}) = 0.4, \]
\[ P(\text{Kangaroo} | \neg \text{Beast}) = 0.0, \]
\( \text{MaternityKangaroo} \equiv \)
\( \text{Kangaroo} \sqcap \exists \text{hasChild. Kangaroo} \)

Inference

\[ P(\text{Parent}(0) | \text{Human}(0)) = 0.232 \]
Goal
Find a correct concept with respect to given examples. A sound concept definition for Target must cover all positive examples and none of the negative examples.
Goal

Find a correct concept with respect to given examples. A sound concept definition for Target must cover all positive examples and none of the negative examples.

- YINYANG
Goal
Find a correct concept with respect to given examples. A sound concept definition for Target must cover all positive examples and none of the negative examples

- YINYANG
- DL-FOIL
Goal

Find a correct concept with respect to given examples. A sound concept definition for Target must cover all positive examples and none of the negative examples

- YINYANG
- DL-FOIL
- DL-Learner
Learning Steps

Refinement Operators

Generalization and specialization $\rightarrow \theta$-subsumption
Learning Steps

Refinement Operators
Generalization and specialization $\rightarrow \theta$-subsumption

Score Function
$\mathcal{K} \cup C \models e$ (instance checking)
Learning Steps

Refinement Operators
Generalization and specialization $\rightarrow \theta$-subsumption

Score Function
$\kappa \cup C \models e$ (instance checking)

Search Algorithm
FOIL-based, genetic algorithms, horizontal expansion
Assumption

Deterministic

Father ≡ Male □ hasChild. ⊤
Assumption

Deterministic
Father ≡ Male ∨ hasChild. ⊤

Probabilistic
\[ P(\text{FlyingBird} | \text{Bird}) = \alpha \]
Assumption

Deterministic
Father ≡ Male ⊓ hasChild. ⊤

Probabilistic
\[ P(\text{FlyingBird} | \text{Bird}) = \alpha \]

Assumption
Negative and positive examples underlie the choice of either a concept definition or a probabilistic inclusion
Proposal

- We expect to find concepts covering all positive examples which is not always possible → uncertainty
Proposal

- We expect to find concepts covering all positive examples which is not always possible → uncertainty
- When we are unable to find a concept definition that covers all positive examples we assume such hypothesis as candidates to be a probabilistic inclusion
Proposal II

**Refinement Operators**

Previous refinement operators
Proposal II

Refinement Operators

Previous refinement operators

Probabilistic score function

\[ \text{cover}(e, \mathcal{K}, C) = P(e|\mathcal{K}, C). \]
Proposal II

Refinement Operators

Previous refinement operators

Probabilistic score function

\[ \text{cover}(e, \mathcal{K}, C) = P(e|\mathcal{K}, C). \]

Search

- We start by searching a deterministic concept
Proposal II

Refinement Operators

Previous refinement operators

Probabilistic score function

\[ \text{cover}(e, \mathcal{K}, C) = P(e|\mathcal{K}, C). \]

Search

- We start by searching a deterministic concept
- If after a set of iterations the score of the best candidate is below a given threshold, a search for a probabilistic inclusion is started
The Algorithm

Require: an initial knowledge base \( \mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle \) and a training set \( \mathcal{E} \).

1: SearchTree with a node \( \{C = \top, h = 0\} \)
2: repeat
3: choose node \( N = \{C, h\} \) with highest probabilistic score in SearchTree
4: expand node to length \( h + 1 \):
5: add all nodes \( D \in (\text{refinementOperator}(C)) \) with length \( =h + 1 \)
6: learn parameters for all nodes \( D \)
7: \( N = \{C, h + 1\} \)
8: expand alternative nodes according to horizontal expansion factor and \( h + 1 \)[18]
9: until stopping criterion
10: \( N' = \text{best node in SearchTree} \)
11: if score(\( N' \)) > threshold then
12: return deterministic concept \( C' \in N' \)
13: else
14: call ProbabilisticInclusion(SearchTree)
15: end if

Algorithm 1: Algorithm for learning probabilistic terminologies.
Probabilistic Inclusion Algorithm

**Require:** SearchTree previously computed
1: **for** each pair of candidates $C_i, C_j$ in first $k$ nodes of the SearchTree **do**
2:   compute the conditional mutual information $I(C_i, C_j|T)$
3: **end for**
4: build an undirected graph in which vertices are the $k$ candidates
5: annotate the weight of an edge connecting $C_i$ to $C_j$ by the $I(C_i, C_j|T)$
6: build a maximum weight spanning tree from this graph
7: add T as parent for each $C_i$
8: learn probabilities for $P(C_i|Parents(C_i))$
9: return the highest probabilistic inclusion $P(T|C') = \alpha$

**Algorithm 2:** Algorithm for learning probabilistic inclusions.
# Description Logic Learning Results

## Table: Description logic learning results

<table>
<thead>
<tr>
<th>Problem</th>
<th>Axioms, Examples</th>
<th>DL-learner correct (length)</th>
<th>Combined approach correct (length)</th>
</tr>
</thead>
<tbody>
<tr>
<td>trains</td>
<td>252,10</td>
<td>100% (5)</td>
<td>100% (5)</td>
</tr>
<tr>
<td>arches</td>
<td>47,5</td>
<td>100% (9)</td>
<td>100% (10)</td>
</tr>
<tr>
<td>moral</td>
<td>31,43</td>
<td>100% (3)</td>
<td>100% (5)</td>
</tr>
<tr>
<td>poker (pair)</td>
<td>35,49</td>
<td>100% (8)</td>
<td>100% (8)</td>
</tr>
<tr>
<td>poker (straight)</td>
<td>45,55</td>
<td>100% (5)</td>
<td>100% (5)</td>
</tr>
</tbody>
</table>
Wikipedia

Wikipedia articles consist mostly of free text, but also contain various types of structured information in the form of Wiki markup
Wikipedia

Wikipedia articles consist mostly of free text, but also contain various types of structured information in the form of Wiki markup

Several Projects

DBPedia and YAGO
YAGO knows more than 2 million entities (like persons, organizations, cities, etc.)
YAGO knows more than 2 million entities (like persons, organizations, cities, etc.)

It knows 20 million facts about these entities: actedIn ...
YAGO knows more than 2 million entities (like persons, organizations, cities, etc.)

It knows 20 million facts about these entities: actedIn...

Scientists and film directors domains
2008 potential scientists have been considered

\[
\begin{align*}
P(\text{wrote}) & = 0.4 \\
P(\text{has Academic Advisor}) & = 0.80 \\
P(\text{interested in}) & = 0.6 \\
P(\text{died on year}) & = 0.7 \\
P(\text{has won prize}) & = 0.4 \\
P(\text{works at}) & = 0.85 \\
P(\text{influences}) & = 0.6 
\end{align*}
\]

\[
\text{Scientist} \equiv \text{Person} \\
\quad \exists (\exists \text{has Academic Advisor}. \text{Person} \\
\quad \quad \exists \text{wrote}. \text{Text} \quad \exists \text{works at}. \text{Educational Institution})
\]

\[
P(\text{Influential Scientist} \mid \text{Scientist} \quad \exists \text{influences}. \\
\quad \exists \text{died on year}. \text{Year}) = 0.85
\]

\[
P(\text{Musician} \mid \text{Person} \quad \exists \text{has Academic Advisor}. \exists \text{wrote}. \text{Text}) = 0.1
\]

\[
\text{Honored Scientist} \equiv \text{Scientist} \\
\quad \exists \text{has won prize}. \text{Prize}
\]
$P(\text{Scientist}(0) \mid \text{Person}(0) \land (\exists \text{wrote}. \text{Text}(1) \land \exists \text{hasAcademicAdvisor}. \text{Person}(2))) = 0.5$
Scientists II

\[ P(\text{Scientist}(0) \mid \text{Person}(0) \land (\exists \text{wrote}. \text{Text}(1) \land \exists \text{hasAcademicAdvisor}. \text{Person}(2))) = 0.5 \]

\[ P(\text{Scientist}(0) \mid \text{Person}(0) \land (\exists \text{wrote}. \text{Text}(1) \land \exists \text{hasAcademicAdvisor}. \exists \text{influences}. \text{Person}(3))) = 0.65 \]
Directors

5589 potential directors have been considered

\[ P(\text{isMarriedTo}) = 0.1 \]
\[ P(\text{influences}) = 0.35 \]
\[ P(\text{hasWonPrize}) = 0.28 \]
\[ P(\text{hasChild}) = 0.05 \]
\[ P(\text{diedOnYear}) = 0.5 \]
\[ P(\text{directed}) = 0.8 \]
\[ P(\text{actedIn}) = 0.4 \]

Actor \equiv \text{Person} \land \forall \text{actedIn.Film}

\[ P(\text{Director}) \mid \text{Person} \land (\exists \text{directed.Film} \land \exists \text{influences.} \exists \text{actedIn.Film}) = 0.75 \]

\[ P(\text{FomerActor}) \mid \text{Director} \land \exists \text{actedIn.Film} = 0.6 \]

HonoredDirector \equiv \text{Director} \land \exists \text{hasWonPrize.Prize}

FamilyDirector \equiv \text{Director} \land (\exists \text{isMarriedTo.Director} \lor \exists \text{hasChild.Director})

\[ P(\text{InfluentialDirector}) \mid \text{Director} \land \exists \text{hasWonPrize.Prize} \land \exists \text{influences.} \exists \text{isMarriedTo.Director} = 0.7 \]

\[ P(\text{MostInfluentialDirector}) \mid \text{Director} \land \exists \text{diedOnYear.Year} \land \exists \text{influences.} \exists \text{hasWonPrize.Prize} = 0.8 \]
Preliminary Results

Directors II

\[ P(\text{Actor}(0) \mid \text{Person}(0) \sqcap \exists \text{actedIn}.\text{Film}(1) \sqcap \exists \text{directed}.\text{Film}(2)) = 0.4 \]

\[ P(\text{Director}(0) \mid \text{Person}(0) \sqcap \exists \text{actedIn}.\text{Film}(1) \sqcap \exists \text{directed}.\text{Film}(2)) = 0.55 \]
Directors II

\[ P(\text{Actor}(0)|\text{Person}(0) \sqcap \exists \text{actedIn}.\text{Film}(1) \sqcap \exists \text{directed}.\text{Film}(2)) = 0.4 \]

\[ P(\text{Director}(0)|\text{Person}(0) \sqcap \exists \text{actedIn}.\text{Film}(1) \sqcap \exists \text{directed}.\text{Film}(2)) = 0.55 \]

\[ P(\text{Actor}(0) \mid \text{Person}(0) \sqcap (\exists \text{actedIn}.\text{Film}(1) \sqcap \exists \text{directed}.\text{Film}(2) \sqcap \exists \text{influences}.\text{Person}(3))) = 0.3 \]
Conclusions

- We have produced a combined scheme, where both the deterministic and probabilistic components receive due attention.
Conclusions

- We have produced a combined scheme, where both the deterministic and probabilistic components receive due attention.
- Initially, the search aims at finding deterministic concepts. If the score obtained is below a given threshold, a probabilistic inclusion search is conducted.
Conclusions

- We have produced a combined scheme, where both the deterministic and probabilistic components receive due attention.
- Initially, the search aims at finding deterministic concepts. If the score obtained is below a given threshold, a probabilistic inclusion search is conducted.
- Preliminary results have focused a real-world domain —YAGO ontology based on Wikipedia.
The End

Thank you