Towards Learning Fuzzy DL Inclusion Axioms

F.A. Lisi and U. Straccia

WILF 2011

9th International Workshop on
Fuzzy Logic and Applications

August 29-31, 2011 - Trani, Italy

Objective

Knowledge is inherently

1. structured
   - description in terms of objects and relations between objects
2. incomplete
   - partial description
3. vague
   - imprecise description

in many real-world domains.
We want to learn the conceptual descriptions of a target concept, given

1. the data stored into a database as fuzzy sets
2. the background knowledge about the data described via a standard Ontology language (specifically, OWL 2 QL)

Description Logics: the Logics behind OWL 2

- Description Logics (DLs)
  - Family of KR formalisms for incomplete structured knowledge
  - Decidable fragments of FOL
  - Expressive power depending on the set of constructors
  - Very expressive DLs at the basis of the W3C OWL 2 standard language for ontologies
- Fuzzy DLs: based on Mathematical Fuzzy Logic
  - Theoretical foundation of KR formalisms for vague knowledge
  - Truth of statements is a matter of degree (*score*) measured on an ordered scale ($[0, 1]$)
  - A *fuzzy interpretation* $\mathcal{I}$ maps each basic statement p_i into $[0, 1]$ and is then extended inductively to all statements
  - A *fuzzy set* R over a countable crisp set X is a function $R \colon X \to [0, 1]$

Fuzzy DL-Lite[2]

- DL-Lite[1]
  - DL behind the *OWL 2 QL* profile
  - Especially aimed at data intensive applications
  - Tractable query answering
- Fuzziness with Gödel logic
  - $a \otimes b = \min(a, b)$
  - $a \Rightarrow b = \begin{cases} 1 & \text{if } a = 0 \\ 0 & \text{otherwise} \end{cases}$
- Ontology-based access to a relational database

[1](Calvanese et al., 2006)
[2](Straccia, 2010)

- Information is stored in a *knowledge base* $\mathcal{K} = \langle \mathcal{F}, \mathcal{O}, \mathcal{A} \rangle$ where:
  - $\mathcal{F}$ is a finite set of *facts* of the form
$$R(c_1, \ldots, c_n)[s] \tag{1}$$
  - $\mathcal{O}$ is a finite set of *inclusion axioms* having the form
$$Rl_1 \sqcap \ldots \sqcap Rl_m \sqsubseteq Rr \tag{2}$$
    where $m \geqslant 1$
    - all Rl_i (*left-hand relation*) and Rr (*right-hand relation*) have the same arity
$$\begin{array}{lcl} Rl & \longrightarrow & A \mid R[i_1, i_2] \\ Rr & \longrightarrow & A \mid R[i_1, i_2] \mid \exists R.A \end{array}$$
  - $\mathcal{A}$ is a finite set of *abstraction statements* of the form
$$R \mapsto (c_1, \ldots, c_n)[c_{score}].sql \tag{3}$$
- Information can be retrieved from $\mathcal{K}$ by means of *ranking queries*, i.e. conjunctive queries with a scoring function to rank the answers
$$\begin{array}{lcl} q(\mathbf{x})[s] & \leftarrow & \exists \mathbf{y}\ R_1(\mathbf{z}_1)[s_1], \ldots, R_l(\mathbf{z}_l)[s_l], \\ & & \mathrm{OrderBy}(s = f(s_1, \ldots, s_l, p_1(\mathbf{z}'_1), \ldots, p_h(\mathbf{z}'_h)) \end{array} \tag{4}$$
  where p_i is a fuzzy set
- Implemented in the SoftFacts system (http://www.straccia.info/software/SoftFacts/SoftFacts.html)

Learning Fuzzy DL-Lite Inclusion Axioms

- the *target concept* H is a DL-Lite atomic concept;
- the *background theory* $\mathcal{K}$ is a fuzzy DL-Lite knowledge base $\langle \mathcal{F}, \mathcal{O}, \mathcal{A} \rangle$
- the *training set* $\mathcal{E}$ is a collection of fuzzy DL-Lite like facts of the form (1) and labeled as either positive or negative examples for H. We assume that $\mathcal{F} \cap \mathcal{E} = \emptyset$;
- the *target theory* $\mathcal{H}$ is a set of inclusion axioms of the form

$$B \sqsubseteq H \tag{5}$$

where H is an atomic concept, $B = C_1 \sqcap \ldots \sqcap C_m$, and each concept C_i has syntax

$$C \longrightarrow A \mid \exists R.A \mid \exists R.\top \ . \tag{6}$$

A FOIL-like algorithm

- The *coverage relation* for a concept $C \neq H$

$$\mathcal{I}_{ILP} \models C(t) \text{ iff } \mathcal{K} \cup \mathcal{E} \models C(t)[s] \text{ and } s > 0 \ . \tag{7}$$

- The *confidence degree* of an inclusion axiom is:

$$cf(B \sqsubseteq H) = \frac{\sum_{t \in P} B(t) \Rightarrow H(t)}{|D|} \tag{8}$$

  where

  - $P = \{t \mid \mathcal{I}_{ILP} \models C_i(t) \text{ and } H(t)[s] \in \mathcal{E}^+\}$;
  - $D = \{t \mid \mathcal{I}_{ILP} \models C_i(t) \text{ and } H(t)[s] \in \mathcal{E}\}$;
  - $B(t) \Rightarrow H(t)$ denotes the degree to which the implication holds for the instance t;
  - $B(t) = \min(s_1, \ldots, s_n)$, with $\mathcal{K} \cup \mathcal{E} \models C_i(t)[s_i]$;
  - $H(t) = s$ with $H(t)[s] \in \mathcal{E}$.

- The *information gain* function uses the above formulas

$$\text{Gain}(cf(r'), cf(r)) = p * (log_2 \, cf(r') - log_2 \, cf(r)) \ ,$$

  where p is the number of distinct positive examples covered by the inclusion axiom r that are still covered by r'.

Learning Set of Inclusion Axioms

function FOIL-Learn-Set-of-Axioms(H, $\mathcal{E}^+$, $\mathcal{E}^-$, $\mathcal{K}$): $\mathcal{H}$
begin
1. $\mathcal{H} \leftarrow \emptyset$;
2. **while** $\mathcal{E}^+ \neq \emptyset$ **do**
3. $r \leftarrow$ FOIL-Learn-One-Axiom(H, $\mathcal{E}^+$, $\mathcal{E}^-$, $\mathcal{K}$);
4. $\mathcal{H} \leftarrow \mathcal{H} \cup \{r\}$;
5. $\mathcal{E}^+_r \leftarrow \{e \in \mathcal{E}^+ | \mathcal{K} \cup r \models e\}$;
6. $\mathcal{E}^+ \leftarrow \mathcal{E}^+ \setminus \mathcal{E}^+_r$;
7. **endwhile**
8. **return** $\mathcal{H}$
end

Learning One Inclusion Axiom

function FOIL-Learn-One-Axiom(H, $\mathcal{E}^+$, $\mathcal{E}^-$, $\mathcal{K}$): r
begin
$B(\mathbf{x}) \leftarrow \top$;
$r \leftarrow \{B(\mathbf{x}) \rightarrow H(\mathbf{x})\}$;
$\mathcal{E}^-_r \leftarrow \mathcal{E}^-$;
while $cf(r) < \theta$ **and** $\mathcal{E}^-_r \neq \emptyset$ **do**
    $B_{best}(\mathbf{x}) \leftarrow B(\mathbf{x})$;
    $maxgain \leftarrow 0$;
    foreach $l \in \mathcal{K}$ **do**
        $gain \leftarrow \mathrm{Gain}(cf(B(\mathbf{x}) \wedge l(\mathbf{x}) \rightarrow H(\mathbf{x})), cf(B(\mathbf{x}) \rightarrow H(\mathbf{x})))$;
        if $gain \geqslant maxgain$ **then**
            $maxgain \leftarrow gain$;
            $B_{best}(\mathbf{x}) \leftarrow B(\mathbf{x}) \wedge l(\mathbf{x})$;
        endif
    endforeach
    $r \leftarrow \{B_{best}(\mathbf{x}) \rightarrow H(\mathbf{x})\}$;
    $\mathcal{E}^-_r \leftarrow \mathcal{E}^-_r \setminus \{e \in \mathcal{E}^- | \mathcal{K} \cup r \models e\}$;
endwhile
return r
end

A Refinement Operator

1. Add atomic concept (A)
2. Add complex concept by existential role restriction ($\exists R.\top$)
3. Add complex concept by qualified existential role restriction ($\exists R.A$)
4. Replace atomic concept (A replaced by A' if $A' \sqsubseteq A$)
5. Replace complex concept ($\exists R.A$ replaced by $\exists R.A'$ if $A' \sqsubseteq A$)

Example: Hotel database

HotelTable		
id	rank	noRooms
h1	3	21
h2	5	123
h3	4	95

RoomTable			
id	price	roomType	hotel
r1	60	single	h1
r2	90	double	h1
r3	80	single	h2
r4	120	double	h2
r5	70	single	h3
r6	90	double	h3

Tower
id
t1

Park
id
p1
p2

DistanceTable			
id	from	to	time
d1	h1	t1	10
d2	h2	p1	15
d3	h3	p2	5

Example: Abstraction statements

$Hotel \mapsto (h.id)$. SELECT h.id
FROM HotelTable h

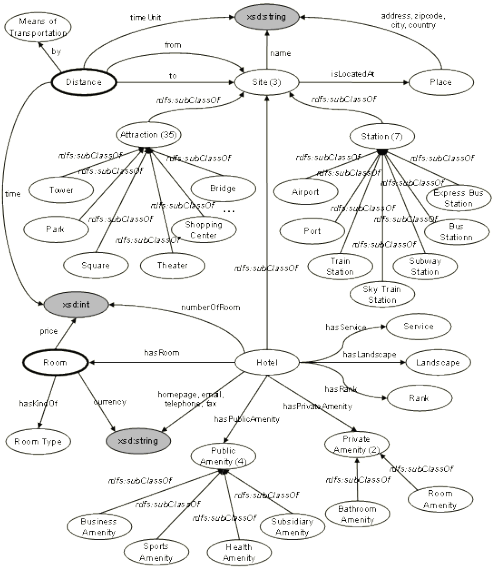
$hasRank \mapsto (h.id, h.rank)$. SELECT h.id, h.rank
FROM HotelTable h

$cheapPrice \mapsto (h.id, r.price)[score]$. SELECT h.id, r.price, *cheap*(r.price) AS score
FROM HotelTable h, RoomTable r
WHERE h.id = r.hotel
ORDER BY score

$closeTo \mapsto (from, to)[score]$. SELECT d.from, d.to *closedistance*(d.time) AS score
FROM DistanceTable d
ORDER BY score

$cheap(p) = leftshoulder(p; 50, 100)$
$closedistance(d) = leftshoulder(d; 5, 25)$

Example: Hotel ontology



$Park \sqsubseteq Attraction$
$Tower \sqsubseteq Attraction$
$Attraction \sqsubseteq Site$
$Hotel \sqsubseteq Site$

Example: Learning

- $H = GoodHotel$
- $\mathcal{E}^+ = \{GoodHotel(h1)[0.6], GoodHotel(h2)[0.8]\}$
- $\mathcal{E}^- = \{GoodHotel(h3)[0.4]\}$.
- $r_0 : \top \sqsubseteq GoodHotel$
  $r_1 : Hotel \sqsubseteq GoodHotel$
  $r_2 : Hotel \sqcap \exists cheapPrice.\top \sqsubseteq GoodHotel$
  $r_3 : Hotel \sqcap \exists cheapPrice.\top \sqcap \exists closeTo.Attraction \sqsubseteq GoodHotel$
  $r_4 : Hotel \sqcap \exists cheapPrice.\top \sqcap \exists closeTo.Park \sqsubseteq GoodHotel$
  $r_5 : Hotel \sqcap \exists cheapPrice.\top \sqcap \exists closeTo.Tower \sqsubseteq GoodHotel$
- Consequence:
  - $cf(r_3) = \frac{0.75 \Rightarrow 0.6 + 0.4 \Rightarrow 0.8}{3} = \frac{0.6+1.0}{3} = 0.5333$.
  - $cf(r_4) = \frac{0.4 \Rightarrow 0.8}{2} = \frac{0.4}{2} = 0.2$.
  - $cf(r_5) = \frac{0.8 \Rightarrow 0.6}{2} = \frac{0.6}{2} = 0.3$.
  - $\mathrm{Gain}(r_4, r_3) = 1 * (log_2(0.2) - log_2(0.5333)) = (-2.3219 + 0.907) = -1.4149$
  - $\mathrm{Gain}(r_5, r_3) = 1 * (log_2(0.3) - log_2(0.5333)) = (-1.7369 + 0.907) = -0.8299$
  - r_5 preferred to r_4 as refinement of r_3
  - r_5 turns out to be consistent w.r.t. $\mathcal{E}$
  - r_5 becomes part of $\mathcal{H}$

Conclusions

- Method for inducing fuzzy DL-Lite inclusion axioms
- Extension of FOIL in a twofold direction
  - from crisp to fuzzy
  - from rules to inclusion axioms

Future work

- To implement and experiment our method
- To analyse the effect of the different implication functions and other parameters in the learning process