Learning Rules on Top of Ontologies: An Inductive Logic Programming Approach

Francesca A. Lisi
lisi@di.uniba.it

LACAM group
Dipartimento di Informatica

Università degli Studi di Bari
Via Orabona, 4 - 70126 Bari - Italy
Overview

- **Motivation**
- Background
- The general framework

- Two instantiations of the general framework
- Conclusions
Motivation: Ontology engineering

- An **ontology** is a formal explicit specification of a shared conceptualization for a domain of interest (Gruber, 1993)
- Description logics are the most used formalisms for specifying ontologies, but ..
- ..Some concepts are better specified by means of **rules**!!

Building rules is a demanding task

Rule induction
**Motivation:**

The Semantic Web

---

**SWRL (Semantic Web Rule Language)**
- submitted to W3C for standardization
- integration of OWL and RuleML
- undecidable

---

- Decidable fragment of SWRL?
- $AL$-log (Donini et al., 1998)!
- Induction for logics?
- Inductive Logic Programming!
Overview

- Motivation
- **Background**
  - The hybrid system $\mathcal{AL}$-log
  - Inductive Logic Programming
- The general framework
- Two instantiations of the general framework
- Conclusions
DL-Horn Hybrid KR&R Systems

- Horn Clausal Logic (HCL) and Description Logics (DLs) are fragments of first-order logic.
- They cannot be compared wrt expressive power but can be combined to obtain more expressive KR&R systems.
  - $\text{AL-log} = \text{ALC} + \text{Datalog}$ (Donini et al., 1998)
  - CARIN (Levy & Rousset, 1998)
The hybrid system $\mathcal{AL}$-log: syntax

$$\mathcal{B} = \langle \Sigma, \Pi \rangle$$

- Equality axioms $C \equiv D$
- Inclusion axioms $C \subseteq D$
- Concept assertions $a : C$
- Role assertions $<a, b> : R$
- Constrained Datalog clauses
  $$\alpha_0 \leftarrow \alpha_1, \ldots, \alpha_m \& \gamma_1, \ldots, \gamma_n$$
  where $\gamma_j$ must be a $\mathcal{ALC}$ concept assertion

- Only one Datalog literal in the head
- Only positive Datalog literals in the body
- Constraints **must** refer to variables occurring in the Datalog part
- Variables in the Datalog part **can** be constrained
The hybrid system \( \mathcal{AL}\)-log: semantics

\[ \mathcal{J} = (I_O, I_H) \]

- \( \mathcal{J} \) satisfies \( \mathcal{B} \) iff
  - it satisfies \( \Sigma \), and
  - for each clause \( \alpha_0 \leftarrow \alpha_1, \ldots, \alpha_m \land \gamma_1, \ldots, \gamma_n \), for each of its ground instances \( \alpha'_0 \leftarrow \alpha'_1, \ldots, \alpha'_m \land \gamma'_1, \ldots, \gamma'_n \), either there exists one \( \gamma'_i, 1 \leq i \leq n \), that is not satisfied by \( \mathcal{J} \) or \( \alpha'_0 \leftarrow \alpha'_1, \ldots, \alpha'_m \) is satisfied by \( \mathcal{J} \)

- OWA of \( \mathcal{ALC} \) and CWA of Datalog do not interfere (safeness)

- UNA holds for \( \mathcal{ALC} \) and ground Datalog
The hybrid system $\mathcal{AL}$-log: reasoning

Consistency of $\Sigma$
- Tableau calculus
  - instance checks ($a:C$ wrt $\Sigma$?)
  - subsumption checks ($C \subseteq D$ wrt $\Sigma$?)

Query answering wrt $\Pi$
- constrained SLD-resolution
  - decidable
  - complete by refutation
    - It tries to obtain the empty constrained Datalog clause (= only constraints) by applying SLD-resolution, then
    - It verifies that constraints in the empty constrained Datalog clause are consistent wrt $\Sigma$ by applying tableau calculus
Inductive Logic Programming

Currently Induction of rules from observations and background knowledge within the representation framework of first-order logic (fragments)
- scope of induction: discrimination/characterization
- task: prediction/description

Originally Induction of rules from observations and background knowledge within the representation framework of Horn clausal logic (Muggleton, 1990)
- scope of induction: discrimination
- task: prediction

Dr. Francesca A. Lisi
Inductive Learning: induction vs deduction

INDUCTION

facts
events
observations

DEDUCTION

theories
rules
models

Dr. Francesca A. Lisi
Inductive Learning: Scope of induction

- **Prediction**

- **Description**
Overview

- Motivation
- Background
- The general framework
  - Problem statement
  - The language of hypotheses
  - The generality order
  - The coverage relations
- Two instantiations of the general framework
- Conclusions
The general framework: problem statement

Goal

Induction of rules from
- a set \( O = \{o_i\} \) of observations, and
- a background knowledge \( K \), \( K \cap O = \emptyset \),
within the representation framework of \( \mathcal{AL}\)-log

General = independent from the scope of induction
The general framework: The language of hypotheses

Hypotheses as constrained Datalog clauses compliant with:

- **linkedness**
  - $h(X) \leftarrow b(X,Y) \land X:C, Y:D \text{ linked}$
  - $h(X) \leftarrow b(X,Y), c(Z) \land X:C, Y:D \text{ not linked}$

- **connectedness**
  - $h(X) \leftarrow b(X,Y) \land X:C, Y:D \text{ connected}$
  - $h(Z) \leftarrow b(X,Y) \land X:C, Y:D \text{ not connected}$

- **Object Identity (OI)**
  - *In a formula terms denoted with different symbols represent different entities of the domain* (Semeraro et al., 1998)
  - Extension of the Unique Names Assumption from the semantics of $\mathcal{ALC}$ to the syntax of $\mathcal{AL}$-log $\rightarrow$ OI-substitution
The general framework: The generality order (1/2)

$\mathcal{B}$-subsumption: A model-theoretic definition

- $\mathcal{B} = K \cup O$
- $H_1, H_2 \in \mathcal{L}$

$H_1$ $\mathcal{B}$-subsumes $H_2$ if
- for every model $J$ of $\mathcal{B}$ and
- every ground atom $\alpha$ such that $H_2$ covers $\alpha$ under $J$,
we have that $H_1$ covers $\alpha$ under $J$.

$H_1$ is at least as general as $H_2$ under $\mathcal{B}$-subsumption, $H_1 \succeq_\mathcal{B} H_2$,
iff $H_1$ $\mathcal{B}$-subsumes $H_2$
The general framework: The generality order (2/2)

$B$-subsumption: A proof-theoretic definition

- $B = K \cup O$
- $H_1, H_2 \in \mathcal{L}$
- $\sigma$ a Skolem substitution for $H_2$ w.r.t. $\{H_1\} \cup B$

$H_1 \geq_B H_2$ iff there exists a substitution $\theta$ for $H_1$ such that
- $\text{head}(H_1)\theta = \text{head}(H_2)$
- $B \cup \text{body}(H_2)\sigma \vdash \text{body}(H_1)\theta\sigma$
- $\text{body}(H_1)\theta\sigma$ is ground.

- Checking $\geq_B$ in $\mathcal{AL}$-log is decidable.
The general framework:
The coverage relations (1/2)

Learning from implications

\[ H = q(X) \leftarrow \text{body}(H) \in \mathcal{L} \]
\[ o_i \text{ is a ground constrained Datalog clause with head } q(a_i) \]

\[ H \in \mathcal{L} \text{ covers } o_i \in O \text{ under entailment w.r.t. } \mathcal{K} \iff \]
\[ \mathcal{K} \cup H \models o_i \]
\[ \mathcal{K} \cup \text{body}(o_i) \cup H \models q(a_i) \]
The general framework: The coverage relations (2/2)

Learning from interpretations

- \( H = \text{q}(X) \leftarrow \text{body}(H) \in \mathcal{L} \)
- \( o_i = (\text{q}(a_i), \mathcal{A}_i) \) where \( \mathcal{A}_i \) is a set of ground Datalog facts

\( H \in \mathcal{L} \) covers \( o_i \in \mathcal{O} \) under interpretations w.r.t. \( \mathcal{K} \) iff
- \( \mathcal{K} \cup \mathcal{A}_i \cup H \models \text{q}(a_i) \)
- \( \mathcal{K} \cup \mathcal{A}_i \cup H \models \text{q}(a_i) \)
The general framework: Conclusions

Related work on learning in hybrid languages
- Frisch, *Sorted downward refinement: Building background knowledge into a refinement operator for ILP*, ILP’99
- Rouveirol and Ventos, *Towards learning in CARIN-ALN*, ILP’00
- Kietz, *Learnability of description logic programs*, ILP’02

Why \(\mathcal{AL}\)-log is to be preferred to CARIN (Levy & Rousset, 1998)
- Safe interaction between DL and HCL part
- Decidable reasoning mechanisms
- Expressive and deductive power *enough* for the actual needs of the Semantic Web

Dr. Francesca A. Lisi
Overview

- Motivation
- Background
- The general framework

- Instantiations of the framework
  - The setting
  - Task 1: Frequent pattern discovery
  - Task 2: Conceptual clustering

- Conclusions
The setting

Characteristic induction from interpretations

- \( \mathcal{L} \) a language of hypotheses
- \( \mathcal{K} \) a background knowledge
- \( \mathcal{M}(\mathcal{B}) \) model constructed from \( \mathcal{B} = \mathcal{K} \cup \mathcal{O} \)
- \( o_i = (q(a_i), \mathcal{A}_i) \) where \( \mathcal{A}_i \) is a set of ground Datalog facts

Finding \( \mathcal{H} \subseteq \mathcal{L} \) such that
- \( \mathcal{H} \) is true in \( \mathcal{M}(\mathcal{B}) \) and
- for each \( \mathcal{H} \in \mathcal{L} \), if \( \mathcal{H} \) is true in \( \mathcal{M}(\mathcal{B}) \) then \( \mathcal{H} \models \mathcal{H} \)
Frequent Pattern Discovery
(at multiple levels of description granularity)

*Given*
- a taxonomic ontology $\Sigma$
- a reference concept $C_{ref} \in \Sigma$
- some task-relevant concepts $C_{tsk}$'s from $\Sigma$
- a data source $\Pi$
- a multi-grained language $\mathcal{L} = \{\mathcal{L}^l\}_{1 \leq l \leq \max G}$ of patterns

*Find* the set of all patterns expressible with $\mathcal{L}$ that describe associations between $C_{ref}$ and $C_{tsk}$'s and are frequent in $r = \Sigma \cup \Pi$
Frequent Pattern Discovery
(at multiple levels of description granularity)

The definition of frequency

Given (also)

- a set \( \{\text{minsup}\} \) of support thresholds
- an evaluation function \( \text{supp} \) for patterns

A pattern \( P \in \mathcal{L}^l \) with \( \text{supp}(P, r) = s \) is frequent in \( r \) iff

- \( s \geq \text{minsup} \)
- all ancestors of \( P \) w.r.t. \( \Sigma \) are frequent in \( r \)

A pattern \( Q \in \mathcal{L}^h, h<l \), is an ancestor of \( P \) iff it can be obtained from \( P \) by replacing each concept \( C \) occurring in \( P \) with a concept \( D \in \mathcal{T}^h \) such that \( C \) is a sub-concept of \( D \)
Frequent Pattern Discovery
(at multiple levels of description granularity)

Find in:
- the on-line CIA World Fact Book (data set CIA) frequent patterns describing:
  - Middle East countries (reference concept MiddleEastCountry) with respect to:
    - the religions believed (task-relevant concepts from the hierarchy rooted in Religion)
      - e.g., the Muslim religion is a monotheistic religion
    - the languages spoken (task-relevant concepts from the hierarchy rooted in Language)
      - e.g., the Indo-Iranian language is an Indo-European language
  at three levels of description granularity (maxG=3)

Dr. Francesca A. Lisi
Instantiating the framework

The data set \( r \)

- A knowledge base in \( \mathcal{AL}\)-log
  \[ \mathcal{B} = < \Sigma, \Pi > \]

An example
- \( \text{MiddleEastCountry} \equiv \text{AsianCountry} \cap \exists \text{Hosts} \cdot \text{MiddleEasternEthnicGroup} \)
- \( \text{AsianCountry} \subseteq \text{Country} \)
- \( 'IR' : \text{AsianCountry}, <'IR','Arab'> : \text{Hosts}, 'Arab' : \text{MiddleEasternEthnicGroup} \)

- \( \text{believes(CountryID, ReligionN)} \leftarrow \text{religion(CountryID, ReligionN, Perc)} \)
  & CountryID:Country, ReligionN: Religion

- \( \text{religion('IR', 'Shia', 89)} \)
Instantiating the framework

The language $\mathcal{L}$ of patterns

- Unary conjunctive queries in $\mathcal{AL}$-log called $O$-queries

$$Q = q(X) \leftarrow \alpha_1, ..., \alpha_m \ & \ X:C_{ref}, \gamma_1, ..., \gamma_n$$

- Generated starting from:
  - A set $\mathcal{A}$ of Datalog predicate names
  - A set $\mathcal{I}$ of $\mathcal{ALC}$ concept names (belonging to the level 1 of description granularity)
Instantiating the framework

The language $\mathcal{L}$ of patterns: an example

$\mathcal{A} = \{\text{believes}/2, \text{speaks}/2\}$

$C_{\text{ref}} = \text{MiddleEastCountry}$

$\Gamma^1 = \{\text{Religion}, \text{Language}\}$

$\Gamma^2 = \{\text{MonotheisticReligion}, \ldots, \text{IndoEuropeanLanguage}\}$

$\Gamma^3 = \{\text{MuslimReligion}, \ldots, \text{IndoIranianLanguage}\}$

$Q_0 = q(X) \leftarrow & X: \text{MiddleEastCountry}$ \hspace{1cm} \text{trivial $\ominus$-query}

$Q_1 = q(X) \leftarrow \text{believes}(X,Y) & X: \text{MiddleEastCountry}, Y: \text{Religion} \in \mathcal{L}^1$

$Q_2 = q(X) \leftarrow \text{believes}(X,Y), \text{speaks}(X,Z) &$

$X: \text{MiddleEastCountry}, Y: \text{MonotheisticReligion}, Z: \text{IndoEuropeanLanguage} \in \mathcal{L}^2$

$Q_3 = q(X) \leftarrow \text{believes}(X,Y), \text{speaks}(X,Z) &$

$X: \text{MiddleEastCountry}, Y: \text{MuslimReligion}, Z: \text{IndoIranianLanguage} \in \mathcal{L}^3$
Instantiating the framework

The evaluation function \( \text{supp} \) for \( \mathcal{L} \)

- It computes the percentage of individuals of \( C_{\text{ref}} \) that satisfy the \( Q \)-query
- It is based on the coverage relation under interpretations

\[
\text{supp}(Q, B) = \frac{| \text{answerset}(Q, B) |}{| \text{answerset}(Q_t, B) |}
\]
Instantiating the framework

The evaluation function $supp$ for $L$: an example

$Q = q(X) \leftarrow \text{believes}(X,Y) \land X:\text{MiddleEastCountry}, Y:\text{JewishReligion}$

$\text{answerset}(Q, \mathcal{B}) = \{\text{IR'}, \text{IL'}, \text{SYR'}\}$

$Q_t = q(X) \leftarrow \& X: \text{MiddleEastCountry}$

$|\text{answerset}(Q_t, \mathcal{B})| = 15$

$supp(Q, \mathcal{B}) = 20\%$

20% Middle East countries believe a Jewish religion
Instantiating the framework

The generality order

\[ \geq_B \text{ is monotone w.r.t support} \]
\[ \text{if } P \geq_B Q \text{ then } \text{supp}(P, B) \geq \text{supp}(Q, B) \]

\[ \geq_B \text{ satisfies all the assumptions underlying Mannila’s levelwise search method!} \]
\[ \text{If a generality order } \geq \text{ over } \mathcal{L} \text{ can be defined such that } \geq \text{ is} \]
\[ \text{monotone w.r.t. supp, the lattice } (\mathcal{L}, \geq) \text{ can be searched with a} \]
\[ \text{breadth-first strategy level by level of depth (Mannila & Toivonen, 1997)} \]

\[ \text{The ILP system } \mathcal{AL-QuIn} \text{ implements this method for} \]
\[ \mathcal{L} \text{ being a language of } \geq_B \text{-ordered } O \text{-queries} \]

Dr. Francesca A. Lisi
Instantiating the framework

The generality order: examples

\[ Q_1 = q(X) \leftarrow \text{believes}(X,Y) \land X:\text{MiddleEastCountry}, Y:\text{Religion} \]
\[ Q_5 = q(X) \leftarrow \text{believes}(X,Y) \land X:\text{MiddleEastCountry}, Y:\text{MonotheisticReligion} \]
\[ Q_6 = q(X) \leftarrow \text{believes}(X,Y), \text{believes}(X,Z) \land X:\text{MiddleEastCountry}, Y:\text{Religion} \]

\[ Q_1 \geq_B Q_5 \] (but not viceversa due to constrained SLD-resolution)
\[ Q_1 \geq_B Q_6 \] (but not viceversa due to the OI bias)
Ontology Refinement

- Adaptation of an existing ontology to a specific domain or the needs of a particular user (Maedche & Staab, 2001)
  - Adding new relations
  - Adding new concepts
Concept Refinement

*Given*

- a taxonomic ontology \( \Sigma \)
- a data source \( \Pi \)
- a concept \( C_{\text{ref}} \in \Sigma \) (reference concept)
- a language \( \mathcal{L} \)

*Find*

a directed acyclic graph (DAG) \( \mathcal{G} \) of concepts \( C_i \) such that:

1. \( \text{int}(C_i) \in \mathcal{L} \)
2. \( \text{ext}(C_i) \subset \text{ext}(C_{\text{ref}}) \)
Concept Refinement (reformulated)

- Concept Refinement $\simeq$ Conceptual Clustering
  - Form of unsupervised learning that aims at determining not only the clusters but also their descriptions in some formalism
Conceptual Clustering: A pattern-based approach (1/2)

- **Assumption:** frequent patterns as clues of data clusters

- **Two-phased method:**
  1. detect *emerging* concepts (known ext, unknown int)
  2. turn emerging concepts into *fully-specified* ones (known ext, known int)

It can rely on an algorithm for frequent pattern discovery, e.g. AL-QuIn!
Different frequent patterns can have the same answer set!

\[ P = q(A) \leftarrow \text{speaks}(A,B), \text{believes}(A,C) \quad \& \quad A: \text{MiddleEastCountry}, B: \text{AfroAsiaticLanguage}, C: \text{MonotheisticReligion} \]

\[ \text{answerset}(P, B) = \{\text{IR'}, \text{SA'}\} \]

\[ \text{ext}(C) = \{\text{IR'}, \text{SA'}\} \]

\[ Q = q(A) \leftarrow \text{speaks}(A,B), \text{believes}(A,C) \quad \& \quad A: \text{MiddleEastCountry}, B: \text{ArabicLanguage}, C: \text{MuslimReligion} \]

\[ \text{answerset}(Q, B) = \text{answerset}(P, B) \]

\[ \text{int}(C) = ? \]
Conceptual Clustering: The choice criterion (1/2)

Language bias

Based on $O$-query form, e.g.
- Minimum level of description granularity
- Nro ontologically constrained variables

Search bias

Based on $B$-subsumption, e.g.
- Most general description
- Most specific description
Conceptual Clustering: The choice criterion (2/2)

Examples

Descriptions must have all the variables ontologically constrained by concepts from the 2nd granularity level on

**m.g.d.**

\[
\text{int}(C) = q(A) \leftarrow \text{speaks}(A,B), \text{believes}(A,C) \land \\
A: \text{MiddleEastCountry}, \quad B: \text{AfroAsiaticLanguage}, \quad C: \text{MonotheisticReligion} \\
\text{ext}(C) = \{\text{IR, SA}\}
\]

**m.s.d.**

\[
\text{int}(C) = q(A) \leftarrow \text{speaks}(A,B), \text{believes}(A,C) \land \\
A: \text{MiddleEastCountry}, \quad B: \text{ArabicLanguage}, \quad C: \text{MuslimReligion} \\
\text{ext}(C) = \{\text{IR, SA}\}
\]
Instantiations of the framework: related work

Frequent pattern discovery

- (Han & Fu, 1999)
  - It can deal with hierarchies
  - It doesn’t adopt FOL

- (Dehaspe & Toivonen, 1999)
  - It can’t deal with hierarchies
  - It does adopt FOL (Datalog)

Conceptual clustering

- Very few works!
  - Some of them do adopt FOL
  - None of them do adopt a pattern-based approach

- (Vrain, 1996)
  - It applies a top-down incremental but distance-based method
  - It does adopt FOL (an object-logical representation)
Bibliography (1/2)


Bibliography (2/2)


Future work

- Study of learnability (i.e. computational complexity of learning) of $\mathcal{AL}$-log
- Extension of the framework towards more expressive hybrid languages (closer to SWRL)
- Instantiation of the framework into cases of discriminant induction (predictive rules)
- More implementation work, e.g. post-processing from $\mathcal{AL}$-log to SWRL
Future work on ResearchCyc

ResearchCyc is the most comprehensive outcome of the Cyc project, began by Douglas Lenat in 1984 and currently carried out by Cycorp, Inc. (Austin, TX, U.S.A.)

The Cyc project's objective was to codify, in machineusable form, the millions of pieces of knowledge that comprise human common sense.

"People have silly reasons why computers don't really think. The answer is we haven't programmed them right; they just don't have much common sense. There's been only one large project to do something about that, that's the famous Cyc project." - Marvin Minsky, MIT, May 2001

"Refining the ResearchCyc ontology with Inductive Logic Programming" is winner of The 2006 Cyc Prize Competition for the best research proposal!
Thanks for your attention!