Machine Learning and Pervasive Computing

Stephan Sigg

15.04.2015

Overview and Structure

13.04.2015	Organisation
13.04.2015	Introduction
20.04.2015	Rule-based learning
27.04.2015	Decision Trees
04.05.2015	A simple Supervised learning algorithm
11.05.2015	–
18.05.2015	Excursion: Avoiding local optima with random search
25.05.2015	–
01.06.2015	k-Nearest Neighbour methods
08.06.2015	High dimensional data
15.06.2015	Artificial Neural Networks
22.06.2015	Probabilistic models
29.06.2015	Topic models
06.07.2015	Unsupervised learning
13.07.2015	Anomaly detection, Online learning, Recom. systems

Outline

Inductive Logic Programming (ILP)

How do we describe our world such that a computer system may take correct decisions from observations?

$\rightarrow$ Inductive reasoning

Inductive Logic Programming (ILP)

ILP can be seen as the reverse of deduction:

Deduction

P_1	All men are mortal
P_2	Socrates is a man
DC	Socrates is mortal

DC <u>must</u> be true

Inductive conclusion

P_1	All men are mortal
P_2	Socrates is mortal
IC	Socrates is a man

IC <u>can</u> be true

Inductive Logic Programming (ILP)

ILP can be seen as the reverse of deduction:

Deduction

P_1	All men are mortal
P_2	Socrates is a man
DC	Socrates is mortal

DC must be true

Inductive conclusion

P_1	All men are mortal
P_2	Socrates is mortal
IC	Socrates is a man

IC can be true

→ In inductive reasoning, starting from facts, we attempt to derive a theory from facts

Inductive Logic Programming (ILP)

ILP can be seen as the reverse of deduction:

Deduction

P_1	All men are mortal
P_2	Socrates is a man
DC	Socrates is mortal

DC must be true

Inductive conclusion

P_1	All men are mortal
P_2	Socrates is mortal
IC	Socrates is a man

IC can be true

→ In inductive reasoning, starting from facts, we attempt to derive a theory from facts

Example

P_1	After rain the ground is wet
P_2	Today, the ground is wet
IC	It rained today

Inductive Logic Programming (ILP)

ILP makes a distinction between three types of knowledge:

- Knowledge
- Observations
- Hypothesis

With increasing count of knowledge and observations, also the possible valid hypothesis rapidly increase

Inductive Logic Programming (ILP)

ILP makes a distinction between three types of knowledge:

Knowledge
Observations
Hypothesis

With increasing count of knowledge and observations, also the possible valid hypothesis rapidly increase

Induction

K	All men are mortal
O	Socrates is mortal
H	Socrates is a man

Induction example

K	Light on = meeting
O	Currently, light is on
H	A meeting is ongoing

Outline

Syntactical – Rule-based learning

Description of Situation by formal Symbols and Rules

Description of a (agreed on?) world view

Pro Combination of rules and identification of loops and impossible conditions feasible

Con

- Complex with more elaborate situations
- Extension or merge of rule sets typically not possible without contradictions

Outline

Rule markup language

Rule ML

- Markup language to express and share rules in XML notation
- Designed for the interchange of various kinds of web-rules

Rule markup language

Language items

<Atom>	Atomic formula
<Implies>	Logical Implication of multiple Atoms
<And>	Logical and
<Or>	Logical or
<Xor>	Logical xor
<Rel>	Relation
<Ind>	Individual constants
<Var>	Variable

Relations can have a variable and finite number of arguments

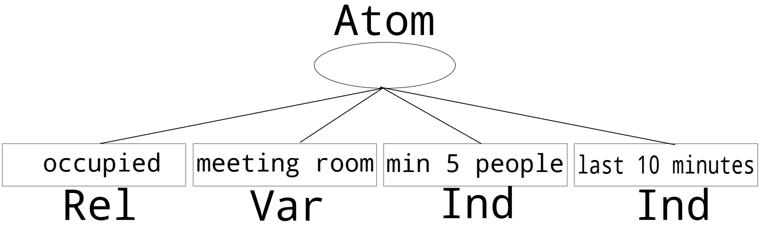
Rule markup language

Rule Markup Language Language for publishing and sharing rules

Hierarchy of rule-sub-languages (XML, RDF, XSLT, OWL)

Example A meeting room was occupied by min 5 people for the last 10 minutes

```
<Atom>
     <Rel> occupied </Rel>
     <Var> meeting room </Var>
     <Ind> min 5 persons </Ind>
     <Ind> last 10 minutes </Ind>
</Atom>
```



Rule markup language

Further conditions with conditional clauses

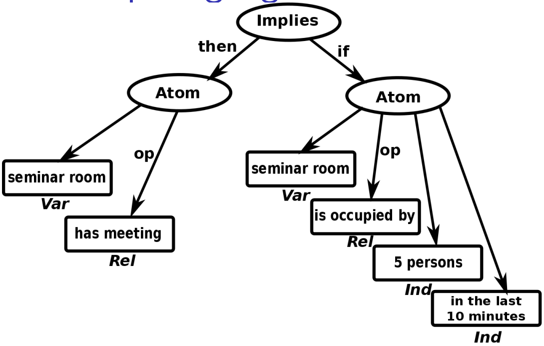
<if> if-then
<then> .

<if> if-do
<do> .

<on> on-do
<do> .

<on> on-if-do
<if>
<do>

Rule markup language




```
<Implies closure="universal">
 <then>
  <Atom>
   <Var>seminar room</Var>
   <Rel>has meeting</Rel>
  </Atom>
 </then>
 <if>
  <Atom>
   <Var>seminar room</Var>
   <Rel>is occupied by</Rel>
   <Ind>5 persons</Ind>
   <Ind>in the last 10 minutes</Ind>
  </Atom>
 </if>
</Implies>
```

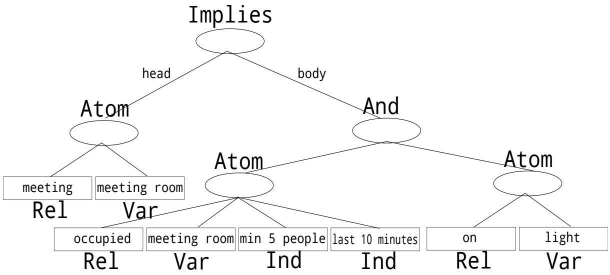
Further conditions with conditional clauses

<if> if-then

<then> .

Rule markup language

- Logical combination of conditions
  - A Meeting is taking place in a meeting room when it was occupied by min 5 people for the last 10 minutes and the light is on.




```
<Implies>
   <head>
      <Atom>
         <Rel> meeting </Rel>
         <Var> meeting room </Var>
      </Atom>
   </head>
   <body>
      <And>
         <Atom>
            <Rel> on </Rel>
            <Var> light </Var>
         </Atom>
         <Atom>
            <Rel> occupied </Rel>
            <Var> meeting room </Var>
            <Ind> min 5 persons </Ind>
            <Ind> last 10 minutes </Ind>
         </Atom>
      </And>
   </body>
</Implies>
```

Rule markup language

Further rules

<Assert> Assertion

<Neg> $\neg$

<Exists> $\exists$

<Forall> $\forall$

<Equal> $=$

$$\neg \exists x : \forall y : x = y$$

```
<Assert>
  <Neg>
    <Exists>
      <Var>x</Var>
      <Forall>
        <Var>y</Var>
        <Equal>
          <Var>x</Var>
          <Var>y</Var>
        </Equal>
      </Forall>
    </Exists>
  </Neg>
</Assert>
```

Ontologies

⇒ Apart from rules, a ground truth is required to describe basic facts

Ontologies

⇒ Apart from rules, a ground truth is required to describe basic facts

Ontologies are a way to describe such ground truth and relation among facts

Ontologies

An ontology is a formal naming and definition of the types, properties, and relationships of fundamental entities in a particular domain

The features of the ontology model should closely resemble the real world

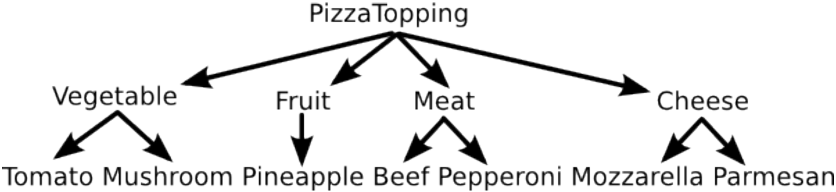
Ontologies

Ontologies – components

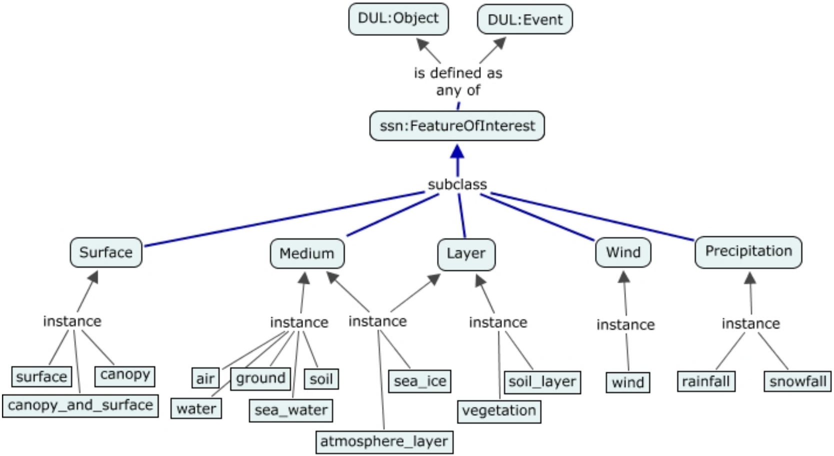
Individuals	instances or objects
Classes	sets, collections, concepts, kinds of things
Attributes	aspects, properties, features, characteristics, or parameters of objects
Relations	between classes and individuals
Function terms	complex structures formed from certain relations
Restrictions	descriptions of what must be true in order for some assertion to be accepted as input
Rules	if-then sentence
Axioms	assertions (including rules) in a logical form
Events	changing of attributes or relations

Ontologies

Ontologies can be created by ontology languages such as the Web Ontology Language (OWL).

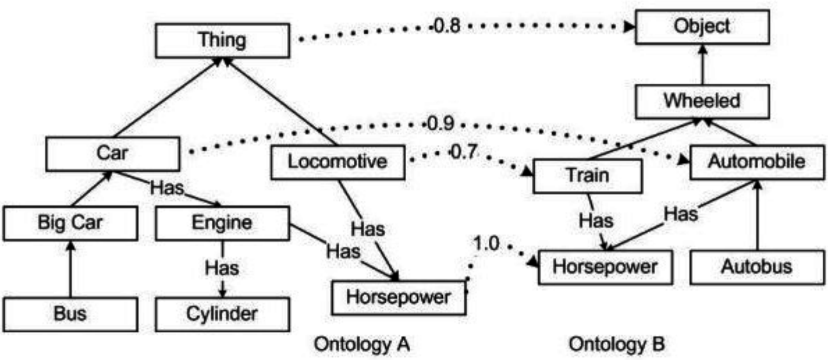


Ontologies


DUL:Object
DUL:Event
is defined as any of
ssn:FeatureOfInterest
subclass
Surface
Medium
Layer
Wind
Precipitation
instance
instance
instance
instance
instance
instance
surface
canopy
canopy_and_surface
air
ground
soil
water
sea_water
sea_ice
atmosphere_layer
soil_layer
vegetation
wind
rainfall
snowfall

Ontologies

Multiple ontologies for the same entity



Semantic Web Rule Language

For further reading: SWRL

SWRL is a Semantic Web Rule Language combining OWL and RuleML

Extends RuleML with ontology axioms:
C(x), P(x,y), sameAs(x,y), differentFrom(x,y), builtIn(r,x,...)

- C: OWL description or data range,
- P: OWL property
- r: built-in relation,
- x and y: variable or OWL individuals or OWL data values

Recognition of patterns

Where Rule-based systems are beneficial

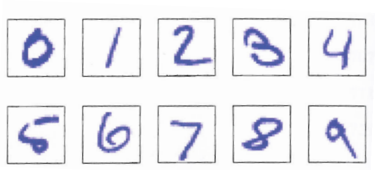
Rule-based systems well suited for limited-size, clearly defined domains with well understood properties
(e.g. detecting high-level activities such as cooking, or other compositions of sequences of short tasks)

Recognition of patterns

Where Rule-based systems are beneficial

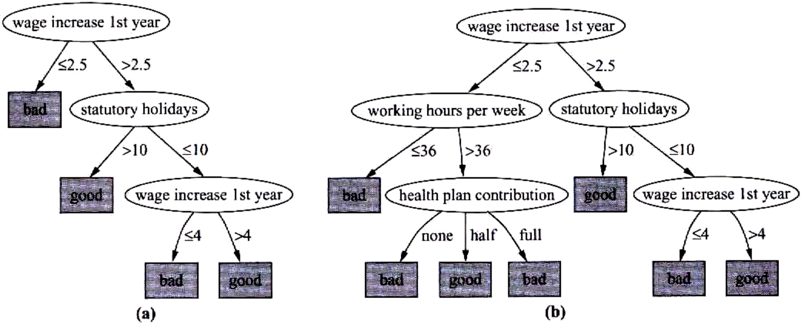
Rule-based systems well suited for limited-size, clearly defined domains with well understood properties
(e.g. detecting high-level activities such as cooking, or other compositions of sequences of short tasks)

Problem: → large problem spaces (might require tremendous amount of rules)
→ Noisy input (e.g. from sensors) requires approximative boundaries



Simple machine learning scheme utilising rules

Decision tree classifier



tbc next week ...

Outline

Inductive Logic Programming

Rule-based learning

RuleML

Questions?

Stephan Sigg
stephan.sigg@cs.uni-goettingen.de

Literature

- C.M. Bishop: Pattern recognition and machine learning, Springer, 2007.
- R.O. Duda, P.E. Hart, D.G. Stork: Pattern Classification, Wiley, 2001.