Fuzzy ILP classifier for Weka

Jan Dědek

Department of Software Engineering, Faculty of Mathematics and Physics,
Charles University in Prague, Czech Republic

Tools Demo Session,
WDS, 2nd June 2010, MFF UK, Troja, Prague 8

http://www.ksi.mff.cuni.cz/~dedek/fuzzyILP/

1. **Fuzzy ILP**
   - Introd. example, theory, architecture and an experiment
   - Fuzzy ILP Implementation
   - Evaluation

ILP Example

Types of ground variables

```
animal(dog). animal(dolphin) ... animal(penguin).
class(mammal). class(fish). class(reptile). class(bird).
covering(hair). covering(none). covering(scales).
habitat(land). habitat(water). habitat(air).
```

Background knowledge

```
has_covering(dog, hair). has_covering(crocodile, scales).
has_legs(dog,4). ... has_legs(penguin, 2). etc.
has_milk(dog). ... has_milk(platypus). etc.
homeothermic(dog). ... homeothermic(penguin). etc.
habitat(dog, land). ... habitat(penguin, water). etc.
has_eggs(platypus). ... has_eggs(eagle). etc.
has_gills(trout). ... has_gills(eel). etc.
```

ILP Example

Positive examples

```
class(lizard, reptile).
class(trout, fish).
class(bat, mammal).
```

Negative examples

```
class(trout, mammal).
class(herring, mammal).
class(platypus, reptile).
```

Induced rules

```
class(A,reptile) :- has_covering(A,scales),
                    has_legs(A,4).
class(A,mammal) :- homeothermic(A), has_milk(A).
class(A,fish) :- has_legs(A,0), has_eggs(A).
class(A,reptile) :- has_covering(A,scales),
                    habitat(A,land).
class(A,bird) :- has_covering(A,feathers).
```

Classical ILP and Fuzzy ILP principles

- Learning examples $E = P \cup N$ (Positive and Negative)
- Background knowledge B
- ILP task – to find hypothesis H such that:

$$(\forall e \in P)(B \cup H \models e) \;\; \& \;\; (\forall n \in N)(B \cup H \not\models n).$$

- Fuzzy learning examples $\mathcal{E} : E \longrightarrow [0, 1]$
- Fuzzy background knowledge $\mathcal{B} : B \longrightarrow [0, 1]$
- Fuzzy ILP task – to find hyp. $\mathcal{H} : H \longrightarrow [0, 1]$ such that:

$$(\forall e_1, e_2 \in E)(\forall \mathcal{M})(\mathcal{M} \models_f \mathcal{B} \cup \mathcal{H}) \; : \; \mathcal{E}(e_1) > \mathcal{E}(e_2) \Rightarrow \|e_1\|_{\mathcal{M}} \geq \|e_2\|_{\mathcal{M}}$$

Generalized Annotated Programs

- Fuzzy ILP is equivalent to Induction of Generalized Annotated Programs[1]
- For implementation we use GAP or strictly speaking: *Definite Logic Programs with monotonicity axioms* (also equivalent)
- Basic paradigm: deal with values as with degrees.
  - We don't have to normalize values, they order is enough.
- For example with monotonicity axioms we can use rule:

  `serious(A, 4)` ← `fatalities(A, 10).`

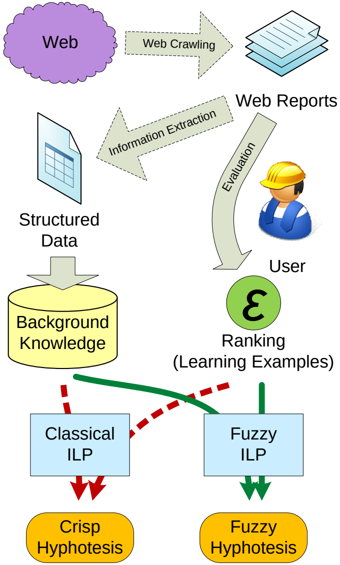
  and from the fact `fatalities(id_123, 1000)` deduce

  `serious_alt(id_123, 4).`

---

[1] See in S. Krajci, R. Lencses and P. Vojtas: "A comparison of fuzzy and annotated logic programming", Fuzzy Sets and Systems, vol.144, pp.173–192, 2004.

Schema of the whole system



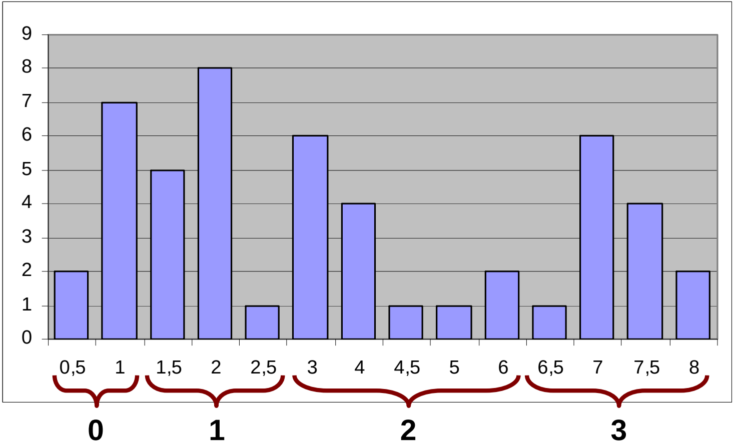
1. Web Crawling
2. Information Extraction and User Evaluation
3. Logic representation
   - Construction of background knowledge
   - Construction of learning examples
4. ILP Learning
   - Crisp
   - Fuzzy
5. Comparison of results

Accident attributes

attribute name	distinct values	missing values	monotonic
size (of file)	49	0	yes
type (of accident)	3	0	no
damage	18	30	yes
dur_minutes	30	17	yes
fatalities	4	0	yes
injuries	5	0	yes
cars	5	0	yes
amateur_units	7	1	yes
profesional_units	6	1	yes
pipes	7	8	yes
lather	3	2	yes
aqualung	3	3	yes
fan	3	2	yes
ranking	14	0	yes

- Information that could be extracted.
- Missing values.
- Almost all attributes are numeric.
  - So monotonic
  - This will be used for "fuzzyfication"
- Artificial target attribute seriousness ranking.

Histogram of the seriousness ranking attribute



- 14 different values, range 0.5 – 8
- Divided into four approximately equipotent groups.

1. **Fuzzy ILP**
   - Introd. example, theory, architecture and an experiment
   - Fuzzy ILP Implementation
   - Evaluation

Essential difference between learning examples

Crisp learning examples

```
serious_2(id_47443). %positive

serious_0(id_47443). %negative
serious_1(id_47443). %negative
serious_3(id_47443). %negative
```

Monotonized learning examples

```
serious_atl_0(id_47443). %positive
serious_atl_1(id_47443). %positive
serious_atl_2(id_47443). %positive

serious_atl_3(id_47443). %negative
```

For one evidence (occurrence):

- Crisp:
  Always one positive and three negative learning examples
- Monotonized:
  Up to the observed degree positive, the rest negative.

Monotonization of attributes

damage_atl ← damage

```
damage_atl(ID,N) :- %unknown values
        damage(ID,N), not(integer(N)).
damage_atl(ID,N) :- %numeric values
        damage(ID,N2), integer(N2),
        damage(N), integer(N), N2>=N.
```

- We infer all lower values as sufficient.
- Treatment of unknown values.
- Negation as failure.

1. **Fuzzy ILP**
   - Introd. example, theory, architecture and an experiment
   - Fuzzy ILP Implementation
   - Evaluation

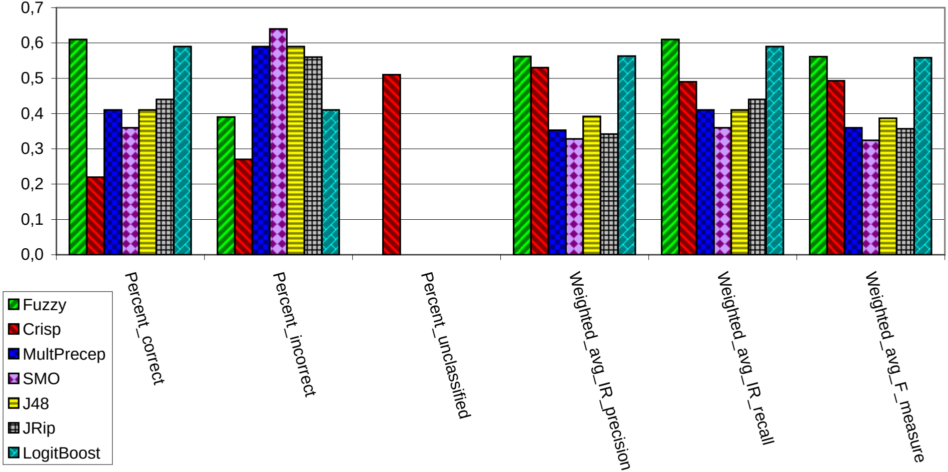
```
serious_0(A):-dur_minutes(A,8).
serious_0(A):-type(A,fire),pipes(A,0).
serious_0(A):-fatalities(A,0),pipes(A,1),lather(A,0).
serious_1(A):-amateur_units(A,1).
serious_1(A):-amateur_units(A,0),pipes(A,2),aqualung(A,1).
serious_1(A):-damage(A,300000).
serious_1(A):-damage(A,unknown),type(A,fire),prof_units(A,1).
serious_1(A):-dur_minutes(A,unknown), fatalities(A,0), cars(A,1).
serious_2(A):-lather(A,unknown).
serious_2(A):-lather(A,0), aqualung(A,1), fan(A,0).
serious_2(A):-amateur_units(A,2),prof_units(A,2).
serious_2(A):-dur_minutes(A,unknown),injuries(A,2).
serious_3(A):-fatalities(A,1).
serious_3(A):-fatalities(A,2).
serious_3(A):-injuries(A,2), cars(A,2).
serious_3(A):-pipes(A,4).
```

- Crisp hypothesis

```
serious_atl_0(A).
serious_atl_1(A):-injuries_atl(A,1).
serious_atl_1(A):-lather_atl(A,1).
serious_atl_1(A):-pipes_atl(A,3).
serious_atl_1(A):-dur_minutes_atl(A,unknown).
serious_atl_1(A):-size_atl(A,764),pipes_atl(A,1).
serious_atl_1(A):-damage_atl(A,8000),amateur_units_atl(A,3).
serious_atl_1(A):-type(A,car_accident).
serious_atl_1(A):-pipes_atl(A,unknown), randomized_order_atl(A,35).
serious_atl_2(A):-pipes_atl(A,3), aqualung_atl(A,1).
serious_atl_2(A):-type(A,car_accident), cars_atl(A,2),prof_units_atl(A,2).
serious_atl_2(A):-injuries_atl(A,1),prof_units_atl(A,3),fan_atl(A,0).
serious_atl_2(A):-type(A,other), aqualung_atl(A,1).
serious_atl_2(A):-dur_minutes_atl(A,59), pipes_atl(A,3).
serious_atl_2(A):-injuries_atl(A,2),cars_atl(A,2).
serious_atl_2(A):-fatalities_atl(A,1).
serious_atl_3(A):-fatalities_atl(A,1).
serious_atl_3(A):-dur_minutes_atl(A,unknown),pipes_atl(A,3).
```

- Monotonized hypothesis
  - Monotonicity axioms
  - Monotonized learning examples

Evaluation and Comparison of Results – graph



	Fuzzy	Crisp	MultPerc	SMO	J48	JRip	LBoost
Corr	0.61±.19	.22±.17 •	.41±.19 •	.36±.24 •	.41±.22 •	.44±.17 •	.59±.26
Incor	.39±.19	.27±.24	.59±.19 ◦	.64±.24 ◦	.59±.22 ◦	.56±.17 ◦	.41±.26
Uncl	.00±.00	.51±.29 ◦	.00±.00	.00±.00	.00±.00	.00±.00	.00±.00
Prec	.56±.24	.53±.37	.35±.20 •	.33±.26	.39±.22	.34±.21 •	.56±.28
Rec	.61±.19	.49±.32	.41±.19 •	.36±.24 •	.41±.22 •	.44±.17 •	.59±.26
F	.56±.20	.49±.33	.36±.19 •	.32±.24 •	.39±.21	.36±.19 •	.56±.27

◦, • statistically significant improvement or degradation

Fuzzy .... czsem.ILP.FuzzyILPClassifier ''

Crisp .... czsem.ILP.CrispILPClassifier ''

MultPerc ...... functions.MultilayerPerceptron '-L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a'

SMO .... functions.SMO '-C 1.0 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1 -K \"functions.supportVector.PolyKernel -C 250007 -E 1.0\"'

J48 .... trees.J48 '-C 0.25 -M 2'

JRip .... rules.JRip '-F 3 -N 2.0 -O 2 -S 1'

LBoost .... meta.LogitBoost '-P 100 -F 0 -R 1 -L -1.7976931348623157E308 -H 0.1 -S 1 -I 10 -W trees.DecisionStump'

Corr .... Percent correct

Inor .... Percent incorrect

Uncl .... Percent unclassified

Prec .... Weighted avg IR precision

Rec .... Weighted avg IR recall

F .... Weighted avg F measure

Conversion of Results

crisp ← monotone (select max)

```
serious_2(ID) :- serious_atl_2(ID),
                 not(serious_atl_3(ID)).
```

monotone ← crisp

```
serious_atl_0(ID) :- serious_2(ID).
serious_atl_1(ID) :- serious_2(ID).
serious_atl_2(ID) :- serious_2(ID).
```