# Towards Ultra-Strong Machine Learning – Comprehensibility of Programs Learned with ILP

Tarek Besold[1], Stephen Muggleton[2], Ute Schmid[3⋆], Alireza Tamaddoni-Nezhad[2], Christina Zeller[3]

[1] University of Bremen, Germany
[2] Imperial College London, UK
[3] University of Bamberg, Germany

In the late 1980s Donald Michie proposed three criteria to evaluate machine learning systems [2]: The *weak criterion* demands that a system improves its performance on unseen data based on learning from a sample of data; the *strong criterion* additionally requires that the system is able to communicate the learned hypotheses in explicit symbolic form; the *ultra-strong criterion* requires furthermore that the user comprehends the system's output and its possible consequences. Standard machine learning only addresses the weak criterion, that is, that learning can be performed with high predictive accuracy. Symbolic, white box learning approaches such as inductive logic programming (ILP, [4]) fulfill Michie's strong criterion. However, it is an open question what types of symbolic hypotheses fulfill the ultra-strong criterion, that is, whether humans are able to comprehend the learned rules, draw the intended consequences and can make use of these rules in their operational context.

As a first step to explore this question, we identified characteristics of logic programs which might affect their comprehensibility:

- the understandability of the predicate names and
- the complexity of the program.

We assume that understandability is higher for 'public' predicate symbols, that is, names related to well-known concepts such as *father(X,Y)* in contrast to 'anonymous' predicate symbols such as *p1(X,Y)*. Complexity of a Prolog program can be characterised by its textual complexity (number of symbols) as well as by its structural complexity (e.g. whether the definition is recursive).

Definitions of predicates in Prolog typically make use of previously introduced predicates. For example, the *grandparent* definition usually is based on *father* and *mother* predicates (see Figure 1) which are used to

⋆ Corresponding author: ute.schmid@uni-bamberg.de

```
; grandparent without invented predicate
p(X,Y) :- father(X,Z), father(Z,Y).
p(X,Y) :- father(X,Z), mother(Z,Y).
p(X,Y) :- mother(X,Z), mother(Z,Y).
p(X,Y) :- mother(X,Z), father(Z,Y).

; grandparent with invented predicate
p(X,Y) :- p1(X,Z), p1(Z,Y).
p1(X,Y) :- father(X,Y).
p1(X,Y) :- mother(X,Y).
```

```
; greatgrandparent without invented predicate
p(X,Y) :- father(X,U), father(U,Z), father(Z,Y).
p(X,Y) :- father(X,U), father(U,Z), mother(Z,Y).
p(X,Y) :- father(X,U), mother(U,Z), father(Z,Y).
p(X,Y) :- father(X,U), mother(U,Z), mother(Z,Y).
p(X,Y) :- mother(X,U), father(U,Z), mother(Z,Y).
p(X,Y) :- mother(X,U), father(U,Z), father(Z,Y).
p(X,Y) :- mother(X,U), mother(U,Z), mother(Z,Y).
p(X,Y) :- mother(X,U), mother(U,Z), father(Z,Y).

; greatgrandparent with invented predicate
p(X,Y) :- p1(X,U), p1(U,Z), p1(Z,Y).
p1(X,Y) :- father(X,Y).
p1(X,Y) :- mother(X,Y).
```

**Fig. 1.** Definition of the grandparent and the greatgrandparent relation without and with an invented predicate *p1* which can be interpreted as parent relation.

introduce a family domain as a list of facts. In addition, further predicates might be introduced to structure Prolog programs – for example, a predicate *parent(X,Y)* which holds if $X$ is either father or mother of $Y$. Some ILP systems allow that such predicates are invented during learning [3].

Hypotheses constructed with such an ILP system which include invented predicates might decrease or increase textual complexity and it might be easy to assign a meaningful name to an invented predicate or not. That is, usage of invented predicates might impact understandability as well as complexity of learned programs. For example, the *grandparent* definition given in Figure 1 has a larger textual complexity without the use of an invented predicate. We assume that if the anonymous symbol *p1(X,Y)* is recognized as representing a parent relation, this second version of grandparent should be easier to understand. For the *greatgrandparent* definition also given in Figure 1, the difference in textual complexity is even more pronounced.

We defined a logic program to be comprehensible if participants showed high accuracy when classifying new material sampled from the same domain. The domain was given as a family tree and participants had to answer questions such as *What is the result of grandparent(mary, jo)?*.

Results indicate that comprehensibility is affected by the textual complexity of the programs and also by the existence of anonymous predicate symbols. Details are given in [1]. For all tested programs, accuracy was significantly higher if participants could correctly name the anonymous predicates. Furthermore, we could partially confirm the hypothesis that predicate invention helps comprehensibility if it decreases textual complexity: We did not find an effect for the grandparent problem where predicate invention reduced complexity by one rule. However, differences

in accuracy were marginally significant for the greatgrandparent problem where predicate invention reduces the number of rules from eight to three.

Since the empirical results are rather promising with respect to Michie's strong criterion, as a next step we are preparing follow-up experiments where the full scenario of explainable machine learning is explored. While in the first experiments, participants were immediately presented with a logic program, in the next experiments, participants first are presented with the learning problem and have to try to formulate a hypothesis. Afterwards, they are presented with a hypothesis learned by the system. Comprehensibility is tested after the first and after the second phase.

White box learning approaches such as ILP and also inductive functional programming [5] in general have a greater chance than standard machine learning approaches to meet Michie's claims and to produce human understandable hyphotheses. Nevertheless, in our opinion empirical research with human participants might help to further improve such symbolic learning approaches with the goal to allow increasingly natural interaction of learning systems and humans.

## References


1. T. Besold, S.H. Muggleton, U. Schmid, A. Tamaddoni-Nezhad, and C. Zeller. How does predicate invention affect human comprehensibility?. In A. Russo and J. Cussens, editors, *Proceedings of the 26th International Conference on Inductive Logic Programming* (ILP 2016, Sept. 4th-6th, London). Springer, Accepted.
2. D. Michie. Machine learning in the next five years. In *Proceedings of the Third European Working Session on Learning*, pages 107–122. Pitman, 1988.
3. S.H. Muggleton, D. Lin, and A. Tamaddoni-Nezhad. Meta-interpretive learning of higher-order dyadic datalog: Predicate invention revisited. *Machine Learning, 100(1)*:4973, 2015.
4. S.H. Muggleton and H. Watanabe, editors. *Latest Advances in Inductive Logic Programming.* Imperial College Press, London, 2015.
5. U. Schmid and E. Kitzelmann. Inductive rule learning on the knowledge level. *Cognitive Systems Research 12(3)*:237-248, 2011.