Optimising Comprehensibility of Machine Learned Logic Programs

Lun Ai

1st supervisor - Prof. Stephen Muggleton; 2nd supervisor - Prof. Alessandra Russo

XAI Background

A recent resurgence of Explainable Artificial Intelligence (XAI) has led to numerous studies and discussions in AI and Machine Learning that seek to ensure understandability. **Common drawbacks in relevant studies** were discussed in multiple survey papers:

- **Under-specified** and **ambiguous definitions**
- **A lack of empirical data** to support claims
- **Limited references to valuable social science literature**
- **No or little accounting for humans' perspective**
- **Not enough emphasis recently on the communicative side**

Aim: investigate and optimise comprehensibility of machine learned logic programs in interactive machine-human teaching contexts.

MIL and predicate invention

Meta-Interpretive Learning (MIL) is a sub-field of Inductive Logic Programming (ILP) which supports predicate invention, dependent learning, learning of recursions and higher-order programs. Given a set of higher-order clauses $\mathcal{M}$ MIL uses logic programming to represent examples by a program $\mathcal{H}$ and background knowledge $\mathcal{B}$.

$$\forall e+ \in E \quad \mathcal{H} \cup \mathcal{B} \cup \mathcal{M} \models e+$$
$$\forall e- \in E \quad \mathcal{H} \cup \mathcal{B} \cup \mathcal{M} \not\models e-$$

In the case background knowledge $\mathcal{B}$ of an ILP system is extended to $\mathcal{B} \cup \mathcal{H}$, we call predicate symbol $p \in P$ an Invention iff p is defined in $\mathcal{H}$ but not in $\mathcal{B}$.

Human comprehension

The operational definition of comprehensibility of logic programs has been taken as the basis for human experimentation and theoretical frameworks to account for explanatory effects. Given a definition D, a group of humans H, a symbolic machine learning algorithm M, the **explanatory effect** $E_{ex}(D, H, M(E))$ of the theory $M(E)$ learned from examples E is

$$E_{ex}(D, H, M(E)) = C_{ex}(D, H, M(E)) - C(D, H, E)$$

The **machine-explained human comprehension** $C_{ex}(D, H, M(E))$ of examples E is the mean accuracy with which a human $h \in H$ after brief study of an explanation based on $M(E)$ can classify new material selected from the domain of D. $C(D, H, E)$ is the unaided human comprehension of examples E. We then relate the explanatory effectiveness of a theory to comprehensibility:

- $M(E)$ learned from examples E is *beneficial* to H if $E_{ex}(D, H, M(E)) > 0$
- $M(E)$ learned from examples E is *harmful* to H if $E_{ex}(D, H, M(E)) < 0$
- $M(E)$ learned from examples E does not have observable effect on H

Due to the analogy between declarative understanding of a logic program and understanding of a natural language explanation, explanatory effects of machine learned theory can be examined after presentation of explanations in the form of English sentences.

Cognitive window

Verbalisation of rules utilises declarative memory and an increase in computational complexity often corresponds to a negative effect on performance of humans. The size of program taught should facilitate human learning and problem solving which has limited working memory capacity. We estimate the mental execution complexity of a query by a new variant of Kolmogorov complexity **cognitive cost** of datalog program Cog and problem solution $CogP$. We hypothesise a bound on human hypothesis space size B based on complexity analysis in and postulated a **cognitive window** which includes two constraints:

1. $E_{ex}(D, H, M(E)) < 0$ if $|S| > B(M(E), H)$
2. $E_{ex}(D, H, M(E)) \leq 0$ if $Cog(M(E), x) \geq CogP(E, \bar{M}, \phi, x)$

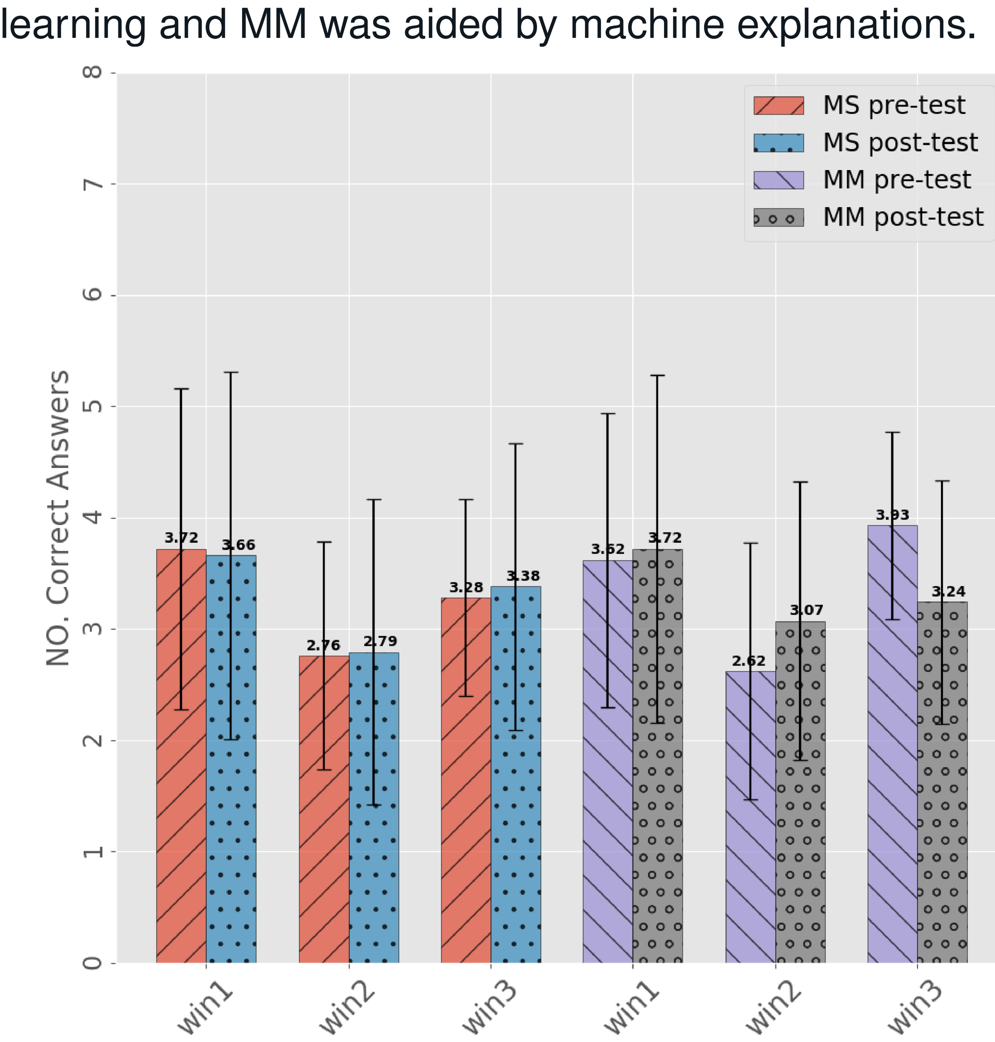
Results

A MIL system $MIPlain$ learns a complete and consistent logic program (below) for tasks win_1, win_2 and win_3 which are tic-tac-toe positions with increasing minimax search depth.

Depth	Rules
1	`win_1(A,B):-move(A,B),won(B)`
2	`win_2(A,B):-move(A,B),win_2_1(B)` `win_2_1(A):-number_of_pairs(A,x,2), number_of_pairs(A,o,0)`

3	`win_3(A,B):-move(A,B),win_3_1(B)`
	`win_3_1(A):-number_of_pairs(A,x,1),win_3_2(A)`
	`win_3_2(A):-move(A,B),win_3_3(B)`
	`win_3_3(A):-number_of_pairs(A,x,0),win_3_4(A)`
	`win_3_4(A):-win_2(A,B),win_2_1(B)`

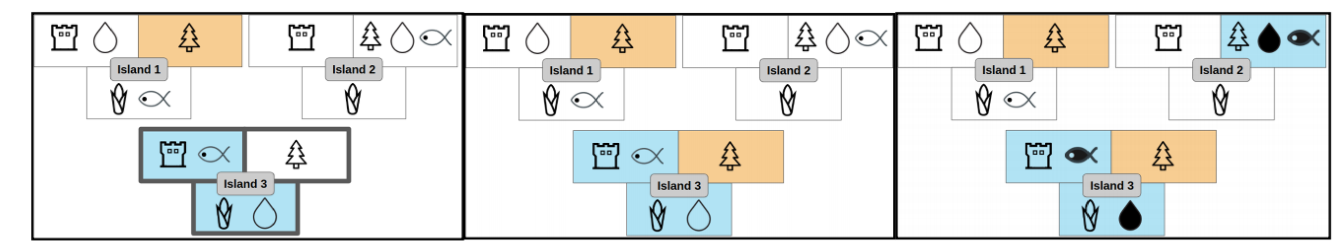
The following diagram shows predictive accuracy for tasks before and after training. MS was human self-learning and MM was aided by machine explanations.



- win_1: violates the cognitive cost constraint and $E_{ex} = 0$
- win_2: does not violate cognitive window constraints and $E_{ex} > 0$
- win_3: violates the hypothesis space size constraint for population that could only remember a maximum of four clauses and $E_{ex} < 0$

Materials

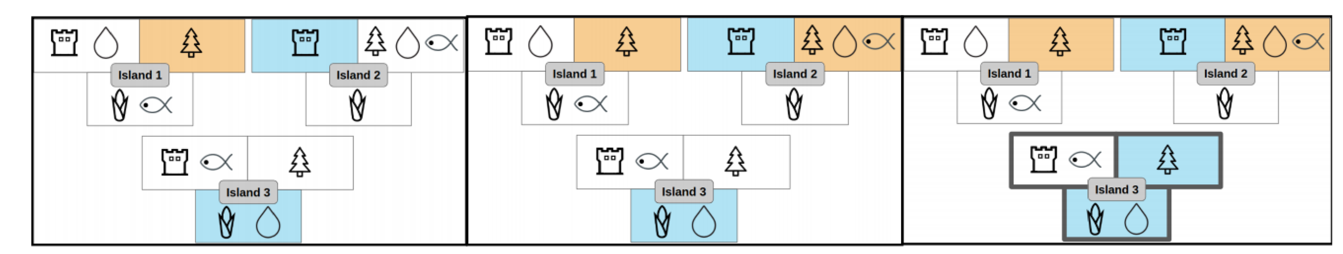
The material is an isomorphism of Noughts and Crosses. Explanations are translated from a machine learned logic theory by $MIPlain$ which is extended based on MIL game learning framework $MIGO$ for winning Noughts and Crosses using additional primitives.



You select this territory and obtain 1 pair (Island 3)

Opponent conquers and prevent you from getting a triplet (Island 3)

You obtain 2 pairs (Water, Fish) and opponent has no pair

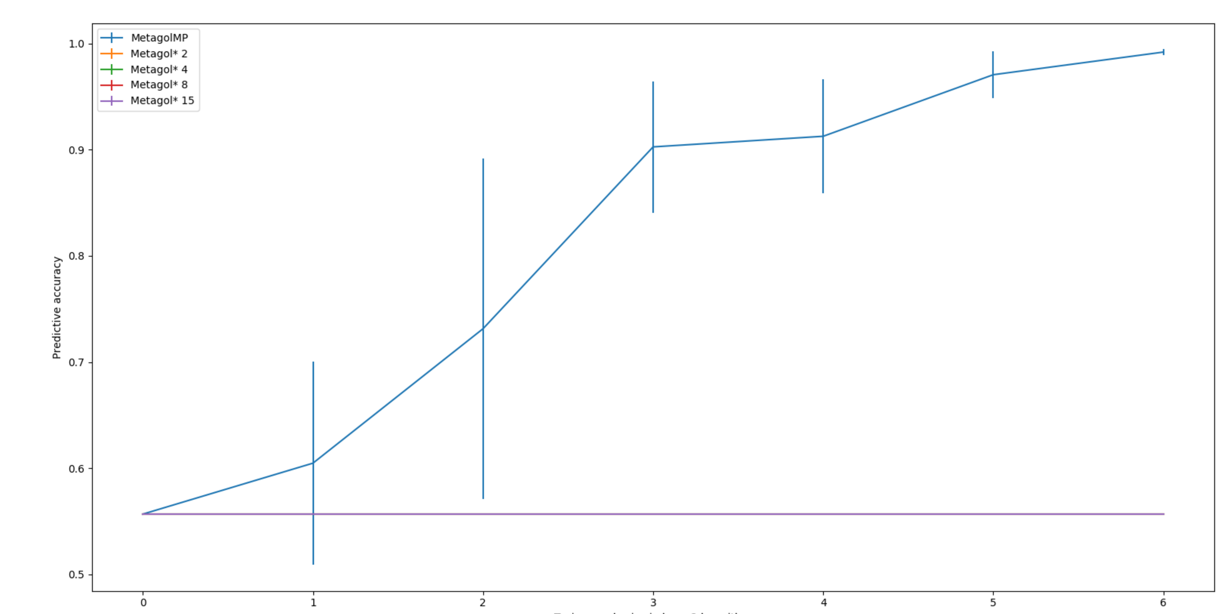


Contrast: Not enough pair(s)

Contrast: Not enough pair(s)

Sequential machine learning of higher-arity recursion

We have developed a new MIL system based on higher-order definite clauses with a head and a body literal of different arities to translate targets of **a higher-arity source language into a lower-arity target language**. Target is projected into simple tasks whose solutions can be learned easily and used for learning recursive definitions from only a smaller set of metarules. Rewrite operator based on resolution and paired projection clauses is both **complete and sound**.



Future and ongoing works

- More interactive human-machine explanatory teaching
- Teaching explanations from stochastic logic programs
- Improving explanatory beneficiality via sequential teaching
- Behavioural debugging of human errors by ILP