

§1 05-13

§1.1 Learning Automata

Dana Angluin (1987)

Model. She assumed the existence of a teacher: minimally adequate teacher.

It's always regular languages. Trying to learn a DFA. You can ask "is w in L ?"

After collecting some information you propose an answer. A teacher says "Yes" or "No" and here is a counter example: a word in L that your proposed machine rejects or a word not in L which your machine incorrectly accepts. Teacher's choice not yours.

Theorem 1.1

With polynomially many queries you are guaranteed to learn the correct unique minimal DFA for L (even if teacher has a different more complex version).

Basic data structure: Observation table. $S, E \subseteq \Sigma^*$, both finite. Helpful to think of S as states and E as experiments.

Assume you have a table with certain properties satisfied. We can propose a DFA

An observation table is said to be closed when

$$\forall t \in S \cdot \Sigma, \exists s \in S \text{ s.t. } \text{row}(t) = \text{row}(s)$$

An observation table is consistent when

$$\forall s_1, s_2 \in S, \text{row}(s_1) = \text{row}(s_2) \Rightarrow \forall a \in \Sigma \text{ row}(s_1, a) = \text{row}(s_2, a)$$

An observation table describes a DFA iff it is closed and consistent.

Algorithm:

1. When it's not closed, you ask for more queries.
2. When it's not consistent, you expand the number of experiments.
3. When it's closed and consistent, you present your DFA. If it's wrong, you use the counter example to expand the table and start over.

Example 1.2

This is an extended example going through the process of finding the DFA. I won't write it down. These **notes** do a great job

A similar algorithm works for weighted automata.

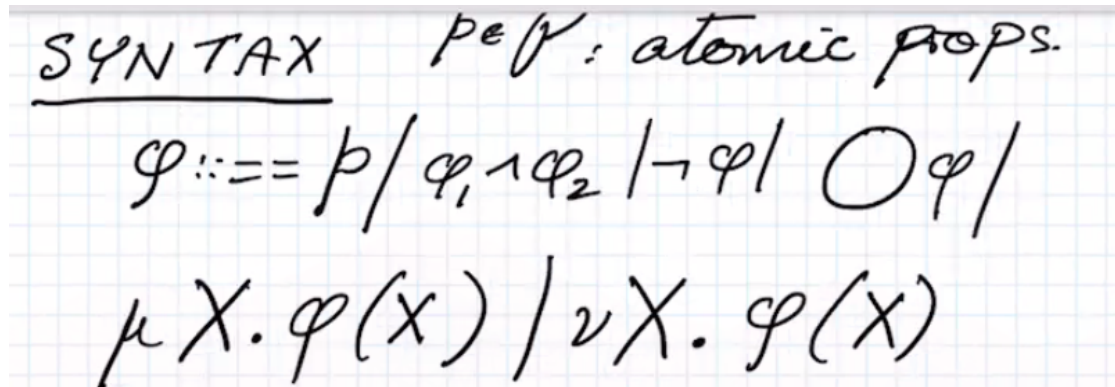
This is kinda automata theory meets machine learning. Another application is in trying to understand what an RNN does. The question of interpreting what an RNN does once it has been trained. It's important to have explainable AI.

Idea: Use an RNN as the teacher and extract a DFA as the explanation of what the RNN is doing. Gail Weiss. Alike Utepoua and Prakash are exploring this.

§1.2 Fixed Point Operators and LTL

$$\Box\varphi = O\varphi \wedge OO\varphi \dots$$

This leads to infinitely many conjunctions. We should be able to prove this with induction.



X is a new syntactic entity. It is a variable that can range over formulas. μX and νX are variable binders. They are called fixed point operators.

Recall 1.3. Every monotone function from a complete lattice to itself has a least fixed point and a greatest fixed point. μ stands for least fixed point. ν stands for greatest fixed point.

We are not allowed to put a fixed point operator binding an X unless X appears in the scope of an even number of negations.

$\mu X. \varphi \wedge \neg X$ and $\mu X. X \Rightarrow \varphi$ are not allowed because they involve negated X . Remember that $x \Rightarrow y$ is the same as $\neg x \vee y$.

We defined the semantics in terms of sequences $\sigma \models \varphi$.

$$\llbracket \varphi \rrbracket = \{ \sigma \mid \sigma \models \varphi \}$$

gives a definable set. Given a formula with a free variable, we interpret it not as a set but as a function from sets to sets.

$$\llbracket \varphi(X) \rrbracket = S \mapsto \llbracket \varphi[S/X] \rrbracket$$

$\varphi[S/X]$ notation for substitute the set S regarded as a formula for X in φ .

e.g. $\varphi(X) = \psi \wedge OX$.

$$\llbracket \varphi(X) \rrbracket(S) = \{ \sigma \mid \sigma \models \psi \wedge \sigma[1..] \in S \}$$

Fact 1.4. With the restriction these functions are always monotone and $P(\Sigma^\omega)$ is a complete lattice.

Now we define

$$\begin{aligned} \llbracket \mu X. \varphi(X) \rrbracket &= lfp \llbracket \varphi(X) \rrbracket \\ \llbracket \nu X. \varphi(X) \rrbracket &= gfp \llbracket \varphi(X) \rrbracket \end{aligned}$$

It's a nightmare to interpret nested fixed points in human terms (although it is still well defined).

$$\Box\varphi = \nu X. \varphi \wedge OX$$

$$\Box\varphi = \nu X. \varphi \wedge OX$$

$$(\Box\varphi)_0 = \text{TRUE}$$

$$(\Box\varphi)_1 = \varphi$$

$$(\Box\varphi)_2 = \varphi \wedge O\varphi$$

$$\begin{aligned} (\Box\varphi)_3 &= \varphi \wedge O(\varphi \wedge O\varphi) \\ &= \varphi \wedge O\varphi \wedge OO\varphi \end{aligned}$$