Relevant Literature

Wesley van der Lee

May 31, 2017

1 Relevant Literature

1.1 Learning Regular Sets from Queries and Counterexamples

Learning Regular Sets from Queries and Counterexamples by Dana Angluin forms the basis of many modern state machine inference algorithms. Her research introduces the polynomial complex L* algorithm for learning a regular set, a task which before was computationally intractable because it was proven to be NP-hard [1]. The basic idea of L* is a learner whose goal is to create a valid conjecture by utilization of an expert system, the Minimally Adequate Teacher (MAT). This is achieved by posing two types of queries to the MAT: membership queries and conjectures. A membership query inquires a given string t to the MAT, who answers *yes* or *no* depending on whether t is a member of the to be hypothesized set U. The learner can also verify a conjecture to the MAT, which answers *yes* if the conjecture is equal to the unknown language. iF this is not the case, the MAT provides a counterexample, which is a string t in the symmetric difference of the conjecture and the unknown language.

The learner keeps track of the queried strings, classified by either a member or non-member of the unknown regular set U. This information is organized in an observation table that consists of three fields: a nonempty finite prefix-closed set S of strings, a nonempty finite suffix-closed set E of strings and a finite function T that maps $((S \cup S \cdot A) \cdot E)$ to $\{0, 1\}$. A word u can be typically of the form $((S \cup S \cdot A) \cdot E)$ and if $u \in U$ then $T(u)$ will result to 1 and 0 otherwise. Thus the observation table can be denoted as (S, E, T).

The learner queries for the right amount of data from the MAT, by establishing an observation table that is closed and consistent. The table is closed if for every t in $S \cdot A$, there exists an s in S such that $row(t) = row(s)$. If the table is not closed, there exists a $t \in S \cdot A$ where no prefix will lead to. The learner identifies for which suffix distinguishes the rows and adds the word to the set S. When $s_1, s_2 \in S$ such that $row(s_1) = row(s_2)$, then if for all a in the alphabet holds $row(s_1 \cdot a) = row(s_2 \cdot a)$, the table is consistent. If the table is not closed, the language would not be regular as it shows non-deterministic behavior. The learner identifies for which $s_1, s_2 \in S$ distinguishes the result of output T and adds the word to the set E.

If (S, E, T) is closed and consistent, a acceptor $M(S, E, T)$ can be defined which is consistent with T and has the smallest set of states. This can be proven as follows. Let q_0 be the starting state $(row(\lambda))$ and δ be the transition function from one state to another in the acceptor, then because $\delta(row(s), a) = row(s \cdot a)$

follows $\forall s \in (S \cup S \cdot A) : \delta(q_0, s) = row(s)$, thus the closed property ensures that any row in the observation table corresponds with a valid path in the acceptor. $\forall s \in (S \cup S \cdot A) \forall e \in E \rightarrow \delta(q_0, s \cdot e)$ is an accepting state if and only if $T(s \cdot e) = 1$, thus due to the consistency with finite function T, a word will be accepted by the acceptor if it is in the regular set. To see that $M(S, E, T)$ is the acceptor with the least states, one must note that any other acceptor M' consistent with T is either isomorphic or equivalent to $M(S, E, T)$ or contains more states. The algorithm L* is listed in Listing 1.

```
S = E = {λ}
(S,E,T) ← MQ(λ) ∪ ∀a ∈ A : MQ(a) # (S,E,T) is the observation table

While M is incorrect: # M is the conjecture
  While (S,E,T) is not consistent or not closed
    if (S,E,T) is not consistent:
        ∃(s1,s2) ∈ S, a ∈ A, e ∈ E : row(s1) = row(s2) and T(s1 · a · e) ≠ T(s2 · a · e)
        E ← a · e
        extend T to (S ∪ S · A) · E using MQ
    if (S,E,T) is not closed:
        ∃s1 ∈ S, a ∈ A : ∀ row(s1 · a) ≠ row(s)
        S ← s1 · a
        extend T to (S ∪ S · A) · E using MQ
  M = M(S,E,T) # (S,E,T) is closed and consistent
  if Teacher replies with a counter−example t:
    add t and all its prefixes to S
    extend T to (S ∪ S · A) · E using MQ
Halt and output M
```

Listing 1: The L* Algorithm

References

[1] E Mark Gold. Complexity of automaton identification from given data. *Information and control*, 37(3):302–320, 1978.