

Some ideas on active learning using membership queries*

Pedro García, Manuel Vázquez de Parga and Damián López
Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
{pgarcia,mvazquez,dlopez}@dsic.upv.es

1 Inference of teams of automata

Three out of four of the algorithms used by our group are based on the inference of a team of automata [1]. Each automata in the team is obtained using a generalized Blue-Fringe inference scheme [2]. Briefly, the generalization consist of: first, a random selection of a *blue* state; second, the random traversal of the *red* states set; if a *red*-mergible state is found, the *blue* state is deterministically merged, otherwise, the *blue* state is promoted to the *red* set; in both cases the *blue* set is updated. This process ends when the *blue* set is empty.

Note that the process is not deterministic and that each run of this scheme may return different automata. Thus, after n iterations of this process we may obtain a set of n automata. This team can be used in several ways to classify test samples. The one we used consist of the processing of the test samples by the automata in the team. If the automata accepts (rejects) the sample, it gives a positive (negative) vote inversely proportional to the square of its size.

2 Maximum discordance criterion

The first algorithm designed by our group uses the above described team inference process.

Briefly speaking, the initial training set is obtained using a small set of the queries available. A team of automata is inferred using this training set. The team is used to classify a set of random generated strings of a certain length. Those strings with higher discordance are selected to query

*Work partially supported by Spanish Ministerio de Educación y Ciencia under project TIN2007-60769

the oracle and added to the training set. This process is iterated while there are queries available.

The discordance value of a string is function of the distance among the absolute value of the vote obtained by the string and the absolute value of the maximum vote that any string can obtain.

In each iteration, in order to select each new set of p queries, a bigger pool of np strings of length k are randomly chosen. If the number of discordant strings in that pool is lower than p , then, the value of k is incremented in one unit, and a new pool of np strings of length is randomly generated. This process is repeated while the number of discordant strings is lower than p . The next iteration considers the final value of k of the previous iteration decremented in one.

3 Canonical generation of L^* experiments

The L^* algorithm by Angluin [3] is the core of the other algorithms our group proposes.

We recall that the algorithm by Angluin uses equivalence queries. This queries guide the experiments to include in the evidence table. The lack of such queries is substituted by the canonical generation of the experiments to include.

Two different traversals (depth-first and breath-first) can be followed to fill in the table. The behaviour of both approaches is quite different, and therefore, both were considered as different algorithms.

Once the limit of queries has been reached, two options were considered: the construction of an automaton from the information on the table; and, the inference of a team of automata using the above described method and the strings queried to the oracle.

The team of automata is inferred using the prefix-tree acceptor of the strings queried to the oracle, where those states detected during the construction of the table are marked as *red* states.

References

- [1] P. García, M. Vázquez de Parga, D. López, and J. Ruiz. Learning automata teams. *LNAI*, 2010. 10th International Colloquium, ICGI-10.
- [2] K. J. Lang, B. A. Pearlmutter, and R. A. Price. Results of the abbadingo one dfa learning competition and a new evidence-driven state merging algorithm. *LNAI*, 1433:1–12, 1998. 4th International Colloquium, ICGI-98.
- [3] D. Angluin. Learning regular sets from queries and counterexamples. *Information and Computation*, 75:87–106, 1987.