A Risk Driven State Merging Algorithm for Learning DFAs

Sandro Spina

Department of Computer Science and AI,
University of Malta

Abstract. When humans efficiently infer complex functions from a relatively few but well-chosen examples, something beyond exhaustive search must probably be at work. Different heuristics are often made use of during this learning process in order to efficiently infer target functions. Our current research focuses on different heuristics through which regular grammars can be efficiently inferred from a minimal amount of examples. A brief introduction to the theory of grammatical inference is given, followed by a brief discussion of the current state of the art in automata learning and methods currently under development which we believe can improve automata learning when using sparse data.

1 Grammatical Inference

A typical definition for learning would be *the act, process, or experience of gaining knowledge.* Within the field of machine learning this process of gaining knowledge is achieved by applying a number of techniques, mainly those relying on heuristic search algorithms, rule-based systems, neural networks and genetic algorithms. This short report focuses on the learning of regular grammars (those languages accepted by finite state machines) by making use of heuristic search algorithms to direct the search. The process of learning grammars from a given set of data is referred to as grammatical inference (GI).

Automata learning is the process of generalizing from a finite set of labelled examples, the language (FSA) which generated them. Let us say that we've got the +ve example set {10, 20, 30, 80}. Positive since these examples are labelled "accepted" by the target language. We can immediately infer that the target language is that of integers divisible by 10 (or rather strings whose length is divisible by 10). However, by overgeneralizing we can also infer that the language is that of even integers (strings whose length is divisible by 2). Both are correct; however as we'll be outlining in the next section, this example illustrates how vital the training sample is (both +ve and -ve samples), for efficient, correct grammatical inference.

The field of grammatical inference finds practical applications within areas such as syntactic pattern recognition, adaptive intelligent agents, computational biology, natural language acquisition and knowledge discovery as illustrated in [6].

In the next section we will be discussing some theoretical background.

2 Preliminaries

Automata learning or identification can be formally expressed as a decision problem.

Given an integer n and two disjoint sets of words D_+ and D_- over a finite alphabet Σ, does there exist a DFA consistent with D_+ and D_- and having a number of states less than or equal to n

The most classical and frequently used paradigm for language learning is that proposed by Gold [3], namely *language identification in the limit*. There are two main variations of this paradigm. In the first one the learner can make use of as much data as necessary. The learning algorithm is supplied with a growing sequence of examples compatible with the target automata. At each step the learner proposes a hypothesis DFA, representing the guessed solution. The algorithm is said to have the *the identification in the limit* property if the hypothesis (consistent with all learning data) remains unchanged for a finite number of guesses. In the second case the number of available learning examples is fixed and the learning algorithm must propose one hypothesis from this set of examples. This algorithm is said to have the *identification in the limit* property if, for any target machine A, it is possible to define a set D_A^r of training examples called the representative sample (characteristic set) of L(A) [4]. Our work currently focuses on this second variation, were we're currently focusing on determining any lower bounds for the sparsity of the training data in order to be able to identify certain classes of regular languages.

Gold [3] has proved that this decision problem is NP-complete, however if the sets D_+ and D_- are somehow representative of the target automaton, there exist a number of algorithms that solve the considered problem in deterministic polynomial time.

In the next section we'll be describing two main GI algorithms.

3 Learning Algorithms

The first algorithm is due to Trakhtenbrot and Barzdin [5]. A uniformly complete data set is required for their algorithm to find the smallest DFA that recognizes the language. Their algorithm was rediscovered by Gold in 1978 and applied to the grammatical inference problem, however in this case uniformly complete samples are not required. A second algorithm, RPNI (Regular Positive and Negative Inference) was proposed by Oncina and Garcia in 1992. Lang [5] proposed another algorithm that behaves exactly in the same way as RPNI during the same year. The RPNI algorithm is based on merging states in the prefix tree acceptor of the sample. Both algorithms are based on searching for equivalent states. These algorithms had a major impact in the field, since now languages of infinite size became learnable. Lang proved empirically that the average case is tractable.

Different control strategies (heuristics) can be adopted to explore the space of DFA constructions. At each step a number of possible merges are possible, thus the merging order of equivalent states determines the correctness of the generated target language hypothesis. To make things clear let us consider the regular expression ab^*a, with $D_+ = \{aba, aa, abbbba\}$ and $D_- = \{b, ab, abbb\}$. The Augmented Prefix Tree Acceptor (APTA) for these training sets is shown in figure 1.

Note that final (accepting) states are labelled **1**, non-final (rejecting) states are labelled **0** and unknown states are marked **?**. The task of the learning algorithms is to determine the correct labelling for the states marked with a **?**. The learning algorithm proceeds by merging states in the APTA, until no more merges are possible.

Rodney Price [2] proposed an evidence driven heuristic for merging states. Essentially this algorithm (EDSM) works as follows :

1. Evaluate all possible pairings of nodes within the APTA

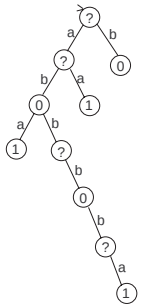

Fig. 1. Augmented Prefix Tree Acceptor

2. Merge the pair of nodes which has the highest calculated evidence score (pair of nodes whose subtrees share the most similar labels).
3. Repeat the steps above until no other nodes within the APTA can be merged.

Figure 2 shows the process of merging states for the APTA shown in figure 1 using the EDSM program available at the Abbadingo web site. The resulting automaton illustrates clearly some of the shortcomings of the EDSM algorithm. Figure 3 illustrates the search space for the learning algorithm. The target automaton lies somewhere between the APTA that maps directly the training data and the universal acceptor.

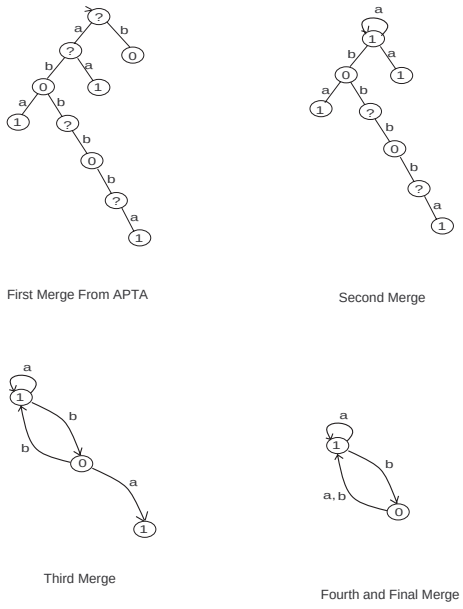

Fig. 2. Execution of EDSM learning algorithm on ab*a

Our current research is devoted at improving these search heuristics. The difficulty of detecting bad merge choices increases as the density of the training data decreases, because the number of labelled nodes decreases within the APTA. In the algorithm we are proposing a *risk* value is associated with each potential merge. During the initial phases of this project we are using various data structures (such as suffix trees) and string algorithms that are helping the algorithm in determining the risk

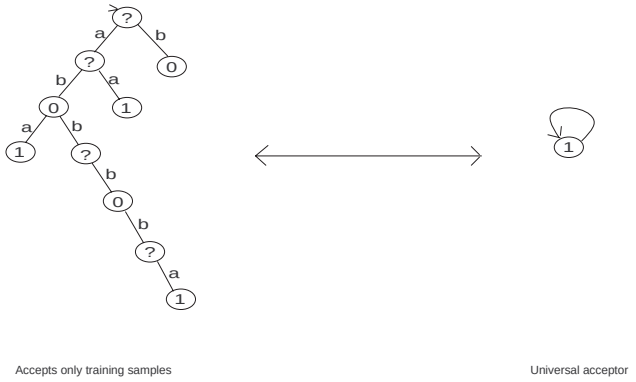

Fig. 3. Search Space for Grammatical Inference

factor for each merge. The algorithm selects the pair of states with the lowest merge risk value and proceeds. Our primary aim is to implement a DFA learning algorithm with variable heuristic parameters that is able to learn target languages from low density sparse training examples.

References

1. Alain Terlutte Francois Denis, Aurelien Lemay. Some classes of regular languages identifiable in the limit from positive data. *Grammatical Inference: Algorithms and Applications. ICGI 2002.*, LNAI 2484:63–76, 2002.
2. Rodney A.Price Kevin J.Lang, Barak A.Pearlmutter. Results of the abbadingo one dfa learning competition and a new evidence-driven state merging algorithm. *Grammatical Inference. ICGI 1998.*, LNAI 1433:1–12, 1998.
3. Stob Osherson and Weinstein. *Systems That Learn: An Introduction to Learning Theory for Cognitive and Computer Scientists.* MIT Press, 1986.
4. L. Miclet P. Dupont and E. Vidal. What is the search space of the regular inference? *Grammatical Inference and Applications. ICGI 1994.*, LNAI 862:25–37, 1994.
5. J. Ruiz P. Garcia, A. Cano. A comparative study of two algorithms for automata identification. *Grammatical Inference: Algorithms and Applications. ICGI 2000.*, LNAI 1891:114–12, 2000.
6. Dana Ron. *Automata Learning and its Applications.* PhD thesis, Hebrew University, 1995.