Learning Commutative Regular Languages. *

Antonio Cano and Gloria Ávarez
(acano,galvarez@dsic.upv.es)
Departamento de Sistemas Informáticos y Computación (DSIC)
Universidad Politécnica de Valencia

September 22, 2008

*Work partially supported by spanish CICYT under TIC2003-09319-C03-02

Introduction

- Regular languages are not inferable from positive samples is a well known result from Angluin (Angluin 1980)
- Some subclasses of Regular Langauges are not inferable from positive samples
- IDEA: Try to improve the behaviour of inference from positive and negative samples of some subclasses of Regular Languages
- Example: k-testable are inferable from positive samples
  Commutative Regular languages are not inferable from positive samples

Outline

- Commutative Regular Languages
- Inference of Commutative Regular Languages from positive samples
- Description of the algorithm ComRPNI (positive and negative samples)
- Experimental Results
- Conclusions

Commutative Regular Languages

Two words $u, v \in \Sigma^*$ are commutatively equivalent if $u = a_1 a_2 \cdots a_n$, and $a_{\sigma(2)} \cdots a_{\sigma(n)} = v$ where σ is a permutation on $\{1, 2, \ldots, n\}$

We denote it by $u \sim_{com} v$.

Examle, $abca \sim_{com} cbaa$.

A language L is commutative if and only if for any $u, v \in \Sigma^*$ such that if $u \in L$ and $u \sim_{com} v$ then $v \in L$.

Examle $L(a + b)^*$ is commutative.

Proposition 1 (Pin) *For every alphabet Σ, the class of commutative languages of Σ is the boolean algebra generated by the languages of the form $K(a, r) = u \in \Sigma^* \mid |u|_a = r$, where $r > 0$ and $a \in \Sigma$, or $L(a, k, p^n) = \{u \in \Sigma^* \mid |u|_a \equiv k \mod p^n\}$, where $0 \leq k < p^n$, p is prime, $n > 0$ and $a \in \Sigma$*

Commutative Regular Languages

- There seems to be some relation between *planar languages* and commutative languages and their inference (Clark et al. 2006).
- Planar Languages are inferable from positive samples and not all planar languages are regular.
- Not all Commutative Regular Languages are Planar Languages .
- An interesting work would be to compare the inference algorithm descrived in (Clark et al. 2006) and the ComRPNI descrived here.

Inference from positive data

Proposition 2 *Commutative regular languages are not inferable from positive samples.*

$F = \bigcup_{n \geq 0} \{a^i \mid i \leq n\} \cup a^*$ is a family of commutative languages which is not inferable from positive data.

$x_1 = a$

$x_2 = aa$

$\ldots$

Commutative deterministic finite automaton (CDFA)

$\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$, where

- $Q = Q_{a_1} \times Q_{a_2} \times \cdots \times Q_{a_n}$,
- $q_0 \in Q$, $F \subseteq Q$
- $\delta((q1, \ldots, q_i, \ldots, q_n), a_i) = (q1, \ldots, \delta_{a_i}(q_i, a_i), \ldots, q_n)$ where δ_{a_i} is a function from Q_{a_i} onto Q_{a_i} for $1 \leq i \leq n$.

Commutative Moore machine $\mathcal{M} = (Q, \Sigma, \Gamma, \delta, q_0, \Phi)$ where

- Σ is an input alphabet
- Γ is an input alphabet
- $Q = Q_{a_1} \times Q_{a_2} \times \cdots \times Q_{a_n}$,
  (where all Q_{a_i} for $1 \leq i \leq n$ are finite sets of states)
- $q_0 \in Q$
- $\delta((q1, \ldots, q_i, \ldots, q_n), a_i) = (q1, \ldots, \delta_{a_i}(q_i, a_i), \ldots, q_n)$ where
  δ_{a_i} is a function from Q_{a_i} onto Q_{a_i} for any $1 \leq i \leq n$
- Φ is a function that maps Q in Γ called *output function*.

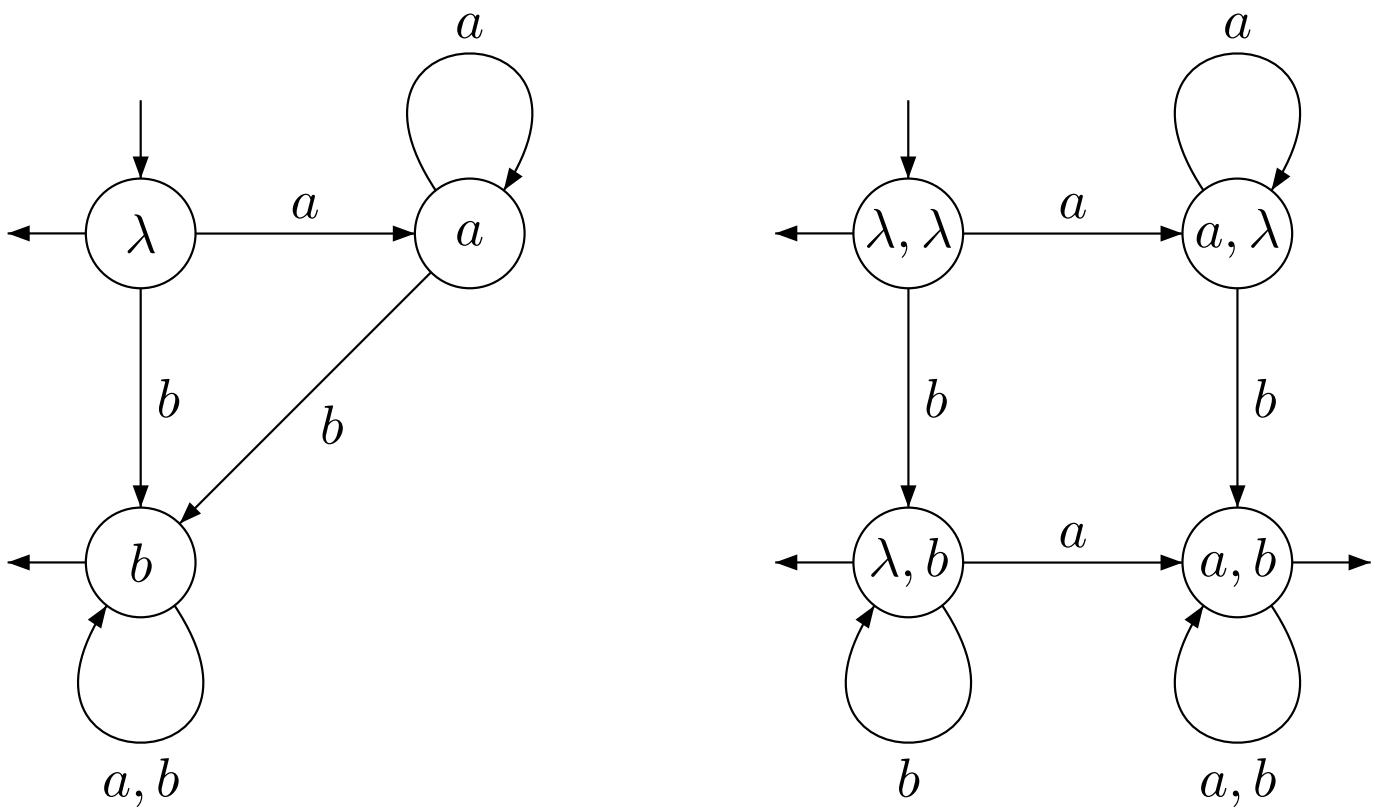

Figure 1: The minimal DFA (on the left) and minimal CDFA (on the right) of the language $\{x \in \Sigma^* \mid |x|_a = 0 \text{ or } |x|_b > 0\}$.

1: $M = CPAM(D_+ \cup D_-)$
2: $\Sigma = alphabet(D_+ \cup D_-)$
3: $listcomp = generateComparisonOrder(M, \Sigma)$
4: **while** $listcomp \neq \emptyset$ **do**
5: $(p_a, q_a) = first(listComp)$ (with $p_a, q_a \in Q_a$ for some $a \in \Sigma$)
6: **while** $\neg emptyset(queue)$ **do**
7: $(p_a, q_a) = pop(queue)$
8: **if** $\neg merge(M, p_a, q_a)$ **then**
9: $M = M'$
10: **end if**
11: **end while**
12: **end while**
13: Return M

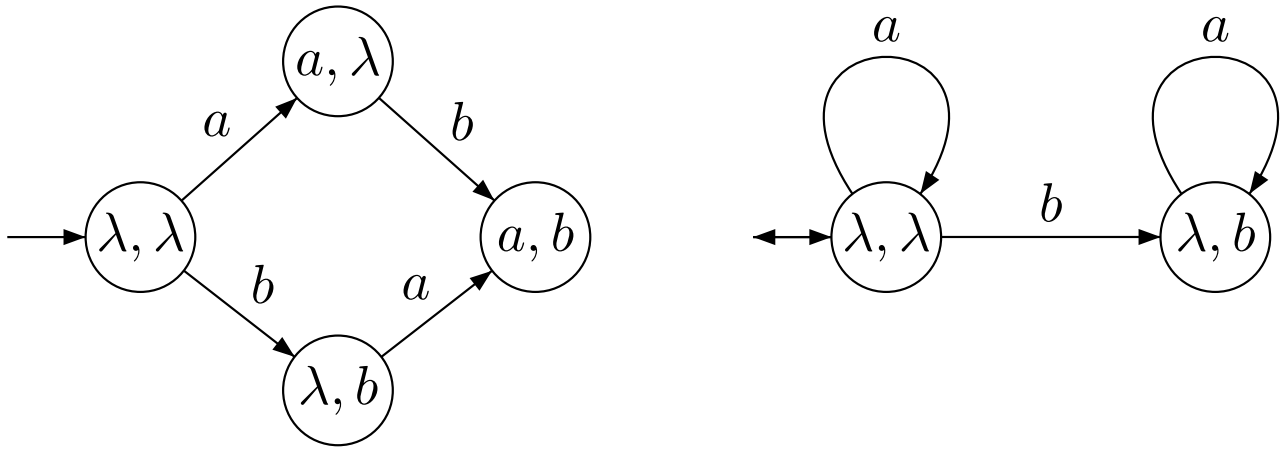

Figure 2: $CPAM(D_+ \cup D_-)$ with $\Phi(\lambda, \lambda) = \uparrow$, $\Phi(a, \lambda) = 1$, $\Phi(\lambda, b) = 0$ and $\Phi(a, b) = \uparrow$, (on the left) and the resulting automaton from the CRPNI algorithm for $D_+ = \{a\}$ and $D_- = \{b\}$ (on the rigth).

Experimental Results

- $|\Sigma| = 3$
- We trained 200 regular commutative regular target languages which states number range between 6 and 90 states.
- The corpus contains incremental training sets of 10, 20, 30, 40, 50, 100, 200, 300, 400 and 500 samples.

Experimental Results: Recognition rates

id	RPNI	*CRPNI*
t10	52.38%	61.67%
t20	52.41%	69.29%
t30	52.85%	77.18%
t40	52.28%	82.96%
t50	52.97%	87.32%
t100	54.06%	96.38%
t200	57.58%	98.84%
t300	58.87%	99.48%
t400	59.80%	99.66%
t500	60.86%	99.77%

Experimental Results: Average state number

id	RPNI	*CRPNI*
t10	4.60	8.09
t20	6.56	17.47
t30	8.31	26.24
t40	9.98	33.01
t50	11.41	36.14
t100	18.24	35.37
t200	28.57	35.69
t300	37.99	34.51
t400	46.11	34.58
t500	54.29	34.62

Conclusions

- We show that Commutative Regular Languages are not inferable from positive data.
- We give a new algorithm that for improving the inference from positive and negative samples.
- We show by an experimentations that this algorithm improves considerably this inference.

Open Problems

- See if this improvement could be applied for real problems.
- Try to find similar algorithm for other subclases of Regular Languages and study the improvement

THANKS

MERCI