A Characterization of Even Linear Languages and Its Application to the Learning Problem*

Jose M. Sempere Pedro García

Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
Camino de Vera s/n, 46071 Valencia (Spain)
email:jsempere@dsic.upv.es email:pgarcia@dsic.upv.es

Abstract

Even Linear Language class is a subclass of context-free class. In this work we propose a characterization of this class using a relation of finite index. Theorems are provided in order to prove the consistence of the characterization. Finally, we propose a method to learn this class using a reduction to the problem of learning regular languages.

1 Introduction.

Formal Language Theory has been applied to learning under the Grammatical Inference paradigm. A survey of this approximation can be found in [2]. Under this paradigm, one way of obtaining good learning algorithms is by providing some characteristics of the formal language class to be learned and by taking advantage of these characteristics to design the algorithms. Typically, the language classes used in grammatical inference have been the context-free class, the regular class or any context-free subclass which could or could not contain the regular class. Some learning algorithms have been proposed to learn some of these classes from information that consisted of given data as strings or skeletons, or different queries.

The Even Linear Language class (ELL) was initially introduced by Amar and Putzolu [1] as a subclass of the more generic Linear Language class. In their work, Amar and Putzolu provided a Nerode-type characterization [4] of the ELL. Under this characterization, a language is even linear iff it is saturated by a finite index quasi-congruence. Informally, a quasi-congruence is similar to a congruence in the sense that given two words, its equivalence implies the equivalence of the words obtained by including the previous words in right and left equal length contexts

Some works have focused on the learning problem of even linear languages. For example, the work done by Radhakrishnan and Nagaraja [6] deals with a finite and positive

*Work partially supported by the Spanish CICYT under grant TIC93-0633-CO2

sample for carrying out the learning task. In their work, they used sample strings to obtain even linear grammars from the structural information that the string skeletons could give in terms of which subskeletons were similar and which were not. This algorithm has been applied to the Picture Description Language (PDL) to recognize simple symmetrical objects as established in the same work. Another study in learning even linear languages is by Takada [7]. In his work, Takada established that every even linear language can be generated by a universal grammar provided with a certain *control set* that regulates the application of its rules. Takada proved that the *control set* of every even linear grammar is a regular language. This allows us to reduce the problem of learning even linear languages to the problem of learning regular languages. With this purpose, the input data are analyzed through the universal grammar and converted into strings of rules from which the *control set* is learned using any regular language inference algorithm. Finally, an even linear grammar can be obtained from the inferred *control set* which generates the same language as the universal grammar with the *control set*.

In the present work, we propose a new characterization of the ELL which allows us to define a canonical grammar associated to an even linear language. This grammar is the minimal size grammar of a set of even linear grammars in a standard form which generates the language and is unique except for those which are isomorphic to it.

The Even Linear Languages learning problem is posed, as in [7], through a reduction to the Regular Languages learning problem. The input data are submitted to a transformation, and a regular language learning algorithm is applied to the transformed data. The inverse transformation provides a hypothesis which consists of an even linear grammar for the input sample, from the inferred automata.

2 Basic concepts and notation.

Let Σ be a finite alphabet and Σ^* the free monoid generated by Σ. For every $x \in \Sigma$, $| x |$ denotes the length of x and λ denotes the string of length zero. Given a language $L \subseteq \Sigma^*$ and $x \in \Sigma^*$, then $x^{-1}L$ and Lx^{-1}, respectively, denote the right quotient and the left quotient of x in L, i.e. $x^{-1}L=\{u \in \Sigma^* \mid xu \in L\}$, $Lx^{-1}=\{u \in \Sigma^* \mid ux \in L\}$.

A finite automaton (FA) over Σ is denoted by a five-tuple $M=(Q,\Sigma,\delta,q_0,F)$, where Q is the set of states, $q_0 \in Q$ the initial state, $F \subseteq Q$ the final states, Σ the input alphabet, and δ the transition function. The language accepted by M is denoted by $L(M)$.

The four tuple $G=(N,\Sigma,P,S)$ denotes a grammar where N and Σ are the nonterminal and the terminal alphabets respectively, P is the set of rules of G and $S \in N$ is the start symbol. $L(G)$ denotes the language generated by G.

An Even Linear Grammar (ELG) is a context-free grammar [4] $G=(N,\Sigma,P,S)$ where all the rules in P are of the following forms

- $A \rightarrow xBy$, where $x,y \in \Sigma^*$, $A,B \in N$ and $| x |=| y |$.
- $A \rightarrow x$, where $x \in \Sigma^*$, $A \in N$.

A language L is an even linear language if there exists an ELG which generates L. The class of Even Linear Languages is a proper subclass of the context-free languages and properly includes the class of regular languages.

Given an ELG, there exists an equivalent ELG where every production is in one of the following standard forms [1]

- $A \rightarrow aBb$, where $a,b \in \Sigma$, $A,B \in N$.
- $A \rightarrow a$, where $a \in \Sigma \cup \{\lambda\}$, $A \in N$.

3 A characterization of the Even Linear Languages.

We are going to propose an alternative characterization that could serve as a base in the learning problem. In the first place, we will establish that given any even linear grammar in the standard form defined above, we can find an equivalent deterministic even linear grammar using a transformation on the strings of the language through the following definition.

Definition 1. Let Σ be an alphabet and let the string $x=a_1a_2...a_{k-1}a_ka_{k+1}...a_{n-1}a_n$, where $\forall\ 1 \leq i \leq n$, $i \neq k$, $a_i \in \Sigma$ and $a_k \in \Sigma \cup \{\lambda\}$. We define the *joined extreme* of x and we denote it by $\sigma(x)$ as the string $a_1a_n \mid a_2a_{n-1} \mid ... \mid a_{k-1}a_{k+1} \mid a_k$. We can define the *joined extreme* of a string in an inductive way through the following two definitions:

$\sigma : \Sigma^* \rightarrow (\Sigma^2 \cup \Sigma)^*$

- $\sigma(a)=a$, $\forall\ a \in \Sigma \cup \{\lambda\}$.
- $\sigma(axb)=ab|\sigma(x)$, $\forall\ a,b \in \Sigma$, $\forall\ x \in \Sigma^*$.

We can extend this definition to languages and provide the *joined extremes* of a language L defined as $\sigma(L) = \{\ \sigma(x) \mid x \in L\}$.

In the same way we can define the inverse transformation as follows

- $\sigma^{-1}(a)=a$, $\forall\ a \in \Sigma \cup \{\lambda\}$.
- $\sigma^{-1}(ab|x)=a\sigma(x)b$, $\forall\ a,b \in \Sigma$, $\forall\ x \in \Sigma^*$.

In this case, $\sigma^{-1}(L) = \{\sigma^{-1}(x) \mid x \in L\}$ and $\sigma^{-1}(\sigma(x))=x$, so $\sigma^{-1}(\sigma(L))=L$.

From the last definition, we can enunciate a theorem which establishes that the transformation σ defined above obtains a regular language from an even linear language, and from this fact, we can define a relation of finite index that characterizes the even linear languages.

Theorem 1 *If $L \subseteq \Sigma^*$ is an even linear language, then $\sigma(L)$ is a regular language*

Proof

Let the language $L=L(G)$, where $G=(N,\Sigma,P,S)$ is an even linear grammar in the standard form. We define the finite automaton $A=(Q,\Sigma',\delta,q_0,F)$, where $Q=N \cup \{q_f\}$, $q_f \notin N$, $\Sigma'=\Sigma^2 \cup \Sigma$, $q_0=S$, $F=\{q_f\}$, δ is defined by the rules:

- If $A \rightarrow aBb \in P$, then $B \in \delta(A,ab)$
- If $A \rightarrow a \in P$, then $\delta(A,a)=\{q_f\}$.

So, through an induction process we can prove that

$$\forall A \in N \; A \Longrightarrow_G^* x \text{ iff } \delta(A,\sigma(x)) \cap F \neq \emptyset.$$

Let us observe that if we take a finite automaton like the one constructed in the previous theorem as input, then we can build an equivalent even linear grammar maintaining the inverse process of the theorem, and the equivalence proof is trivial In this case, given an automaton A, the obtained grammar generates the language $\sigma^{-1}(\mathrm{L}(A))$. In Figure 1, we can observe an example of an even linear grammar and its associated finite automaton.

Once we have shown a correspondence between an even linear language L and its regular transformed language $\sigma(L)$, we can establish certain relationships between the regular language theory and similar results for even linear languages to obtain a relation of finite index which produces an equivalent result to the Myhill-Nerode Theorem [4] In order to do this, we will give another definition.

Definition 2. Given an even linear grammar in the standard form $G=(N,\Sigma,P,S)$, we will say that this grammar is *deterministic* if $A \rightarrow aBb \in P$ and $A \rightarrow aCb \in P$ imply that $C=B$.

Theorem 2 *Given an even linear grammar in the standard form* $\mathrm{G}=(\mathrm{N},\Sigma,\mathrm{P},\mathrm{S})$, *then a deterministic even linear grammar in the standard form* G' *exists such that* $L(\mathrm{G})=L(G')$.

Proof

Given G we can obtain a FA A which accepts $\sigma(\mathrm{L}(G))$ as established in Theorem 1. Using operations on this automaton [4], we can obtain an equivalent deterministic FA A'. Keeping similar rules to those used in the previous theorem, the grammar that we associate to A' accepts $\sigma^{-1}(\sigma(\mathrm{L}(G))=\mathrm{L}(G)$.

Finally, we can establish an equivalence relation taking $(\Sigma \mathrm{x} \Sigma)^*$ as the relationship domain.

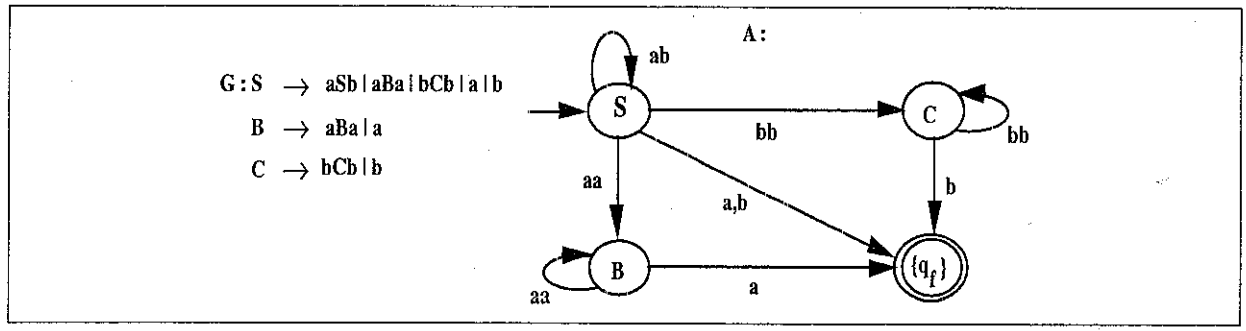

Figure 1: An Even Linear Grammar G and its associated FA A. $\mathrm{L(A)}=\sigma(\mathrm{L(G)})$

Definition 3. Given a language L, we will say that the string pairs (u_1,v_1) and (u_2,v_2) are related (they are *undistinguishable*) under the language L and we denote it by $(u_1,v_1) \equiv_L (u_2,v_2)$ iff

- $\mid u_1 \mid = \mid v_1 \mid, \mid u_2 \mid = \mid v_2 \mid$
- $\forall\, w \in \Sigma^* \; u_1 w v_1 \in L$ *iff* $u_2 w v_2 \in L$ or, using an alternative notation, we will say that $(u_1,v_1)L=(u_2,v_2)L$, where $(u,v)L=u^{-1}(Lv^{-1}) = (u^{-1}L)v^{-1}$.

From the above definition, we can establish the definitive result to characterize the Even Linear Language class. We will do this by the following theorem

Theorem 3 L $\subseteq \Sigma^*$ *is an even linear language iff* $\equiv_L$ *has a finite index*

Proof

- Necessary condition proof.

  Let the language L=L(G), where G=(N,Σ,P,S) is a deterministic even linear grammar as established in Theorem 2. We define the relation $\equiv_G$ over (ΣxΣ)* as follows. $(u_1,v_1) \equiv_G (u_2,v_2)$ *iff*

  - $\mid u_1 \mid = \mid v_1 \mid, \mid u_2 \mid = \mid v_2 \mid$
  - $S \Longrightarrow_G^* u_1Av_1$ *iff* $S \Longrightarrow_G^* u_2Av_2$

  Obviously, $\equiv_G$ has a finite index, given that it will have as many equivalence classes as nonterminal symbols of the grammar. We will prove that if $(u_1,v_1) \equiv_G (u_2,v_2)$, then $(u_1,v_1) \equiv_L (u_2,v_2)$, and therefore $\equiv_L$ has a finite index.

  $(u_1,v_1) \equiv_G (u_2,v_2) \Rightarrow \forall\ w \in \Sigma^*\ u_1wv_1 \in L$ *iff* $u_2wv_2 \in L \Leftrightarrow (u_1,v_1) \equiv_L (u_2,v_2)$.

- Sufficient condition proof.

  Then, let us suppose that $\equiv_L$ has a finite index. Let us define the grammar G=(N,Σ,P,S), where N=$\{(u,v)L \mid u,v \in \Sigma^*$ and $\mid u \mid = \mid v \mid\}$, S=$(\lambda,\lambda)L$ and P is defined through the following rules

  - If $(u,v)L$=A and $(ua,bv)L$=B, then $A \rightarrow aBb \in P$.
  - If $a \in A \cap \{\Sigma \cup \{\lambda\}\}$, then $A \rightarrow a \in P$.

  Then let us see that L(G)=L. In the first place, we could prove that $(u,v)L$=A *iff* $S \Longrightarrow_G^* uAv$, through an induction process.

  Once this has been proved, we can see that L(G)=L, through a double inclusion proof.

  - L(G) $\subseteq L$

    Let us take $x \in$ L(G). Then $S \Longrightarrow_G^* uAv \Longrightarrow_G uav$=$x$ with $\mid u \mid = \mid v \mid$ and $a \in (\Sigma \cup \{\lambda\})$, then $(u,v)L$=A and $a \in A$, so uav=$x \in L$.
  - $L \subseteq$ L(G)

    Let x=$uav \in L$ with $\mid u \mid = \mid v \mid$ and $a \in (\Sigma \cup \{\lambda\})$, then $a \in (u,v)L$=A. So, it is clear that $A \rightarrow a \in P$ and $S \Longrightarrow_G^* uAv$, so $S \Longrightarrow_G^* uAv \Longrightarrow_G uav$=$x \in$ L(G)

Let us see an example of how to construct an even linear grammar from the equivalence relation as established in the previous theorem.

Example

L=aa^*b^*b

$(a,a)L=\oslash$	$(a,a)A=a^*=B$	$(a,a)B=a^*=B$	$(a,a)C=\oslash$
$(a,b)L=a^*b^*=A$	$(a,b)A=a^*b^*=A$	$(a,b)B=\oslash$	$(a,b)C=\oslash$
$(b,a)L=\oslash$	$(b,a)A=\oslash$	$(b,a)B=\oslash$	$(b,a)C=\oslash$
$(b,b)L=\oslash$	$(b,b)A=b^*=C$	$(b,b)B=\oslash$	$(b,b)C=b^*=C$

so, the obtained grammar (maintaining the construction of Theorem 3) will be the following one:

$S \rightarrow aAb$ $\quad A \rightarrow aBa \mid aAb \mid bCb \mid a \mid b \mid \lambda$

$B \rightarrow aBa \mid a \mid \lambda$ $\quad C \rightarrow bCb \mid b \mid \lambda$

Finally we can enunciate a result related to the minimum size of the even linear grammars.

Theorem 4 *The constructed grammar of Theorem 3 is the minimal deterministic grammar which generates* L *and is the only one except for isomorphic ones.*

Proof

As seen in Theorem 3, given a grammar G, the induced relation $\equiv_G$ is a refinenment over the relation $\equiv_L$, so the number of auxiliary symbols induced by $\equiv_G$ is greater than the number of those induced by $\equiv_L$.

4 Application to the learning problem.

Once we have presented a characterization of the class of the even linear languages, our purpose is to apply it to its learning. It can easily be seen that learning an even linear language L can be solved by learning its associated regular language $\sigma(L)$, so the problem of learning even linear languages is obviously resolved. The characterization of the Even Linear Languages proposed in this work is different from the characterization used in [7] but allows us to obtain a result over the learning of the even linear languages which is completely equivalent.

Thus, the scheme to be carried out to learn any even linear language could be the one proposed in Figure 2

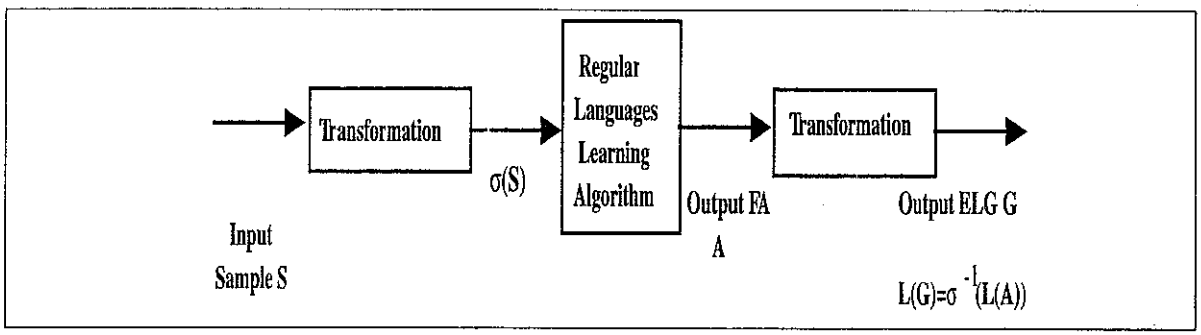

Figure 2: A scheme to learn Even Linear Languages

The proposed scheme is easy to understand. Given a sample of an even linear language, the transformation σ is applied and it obtains a regular language sample. Then, any

regular language learning algorithm can be applied over the transformed sample and, last, the transformation that obtains even linear grammars from finite automata is made by the inverse result of Theorem 1. Let us note that in the module that uses any regular language learning algorithm, an algorithm like the one proposed in [5] could be used which uses a complete presentation sample as input and obtains finite automata as a hypothesis. This algorithm identifies any regular language in the limit [3], so, in such a case, any even linear language can be identified.

Another way of carrying out the identification of any even linear language in the limit could be done by providing algorithms that work without making a reduction of this problem to the regular language identification problem. It could be done by providing a nonterminal merging technique in a similar way of that applied in [5] and [8]. The results of Theorem 3 and Theorem 4 will help to prove the convergence of the algorithm.

5 Acknowledgements

We would like to thank Manuel Vázquez de Parga for his helpful suggestions and his original contribution to the transformation of Even Linear Languages to Regular Languages.

References

[1] V. Amar, G. Putzolu On a Family of Linear Grammars. *Information and Control* 7, pp 283-291. 1964.

[2] D. Angluin, C. Smith Inductive Inference: Theory and Methods. *Computing Surveys* 15, No. 3 pp 237-269. 1983.

[3] M. Gold Language Identification in the Limit. *Information and Control* 10, pp 447-474. 1967.

[4] J. Hopcroft, J. Ullman *Introduction to Automata Theory, Languages and Computation.* Addison-Wesley Publishing Company. 1979.

[5] J. Oncina, P. García Inferring regular languages in polinomial update time. *Pattern Recognition and Image Analysis. Selected Papers from the IVth Spanish Symposium.* Series in Machine Perception Artificial Intelligence. Vol. 1. World Scientific. 1992.

[6] V. Radhakrishnan and G. Nagaraja Inference of Even Linear Grammars and Its Application to Picture Description Languages. *Pattern Recognition* 21, No. 1 pp 55-62. 1988.

[7] Y. Takada Grammatical Inference of Even Linear Languages based on Control Sets. *Information Processing Letters* 28, No. 4 pp 193-199. 1988.

[8] B. Trakhienbrot, Y. Barzdin *Finite Automata. Behavior and Synthesis.* North Holland Publishing Company. 1973.