Vol. VIII, Nº 1, Mayo (2000)
Matemáticas: 1–15

Matemáticas:
Enseñanza Universitaria


# Acerca de los modelos de lenguaje basados en gramáticas estocásticas

Fredy Amaya Robayo

**Resumen**

Los modelos de lenguaje (ML) son modelos matemáticos utilizados como componentes importantes en aplicaciones computacionales, tales como reconocimiento del habla, traducción automática, reconocimiento óptico de caracteres, recuperación de la información, etc. Los modelos de lenguaje estocásticos (MLE) han ganado considerable aceptación debido a la eficiencia demostrada en los campos en que han sido utilizados. El modelo de lenguaje estocástico más utilizado es el modelo de $n$-gramas, el cual está basado en la frecuencia de aparición de una cadena del lenguaje dentro de un gran texto denominado corpus. A pesar que el modelo de $n$-gramas es fácil de implementar y de gran poder expresivo; en los casos en los que se aplica al lenguaje natural o a una estructura lingüística compleja, hay información de la cadena no contenida en las $n$ últimas palabras que el modelo de $n$-gramas no puede captar.
Una alternativa en tales casos la representan los modelos basados en gramáticas libres de contexto, pero, cuando la tarea a realizar es de gran complejidad, se presentan dificultades computacionales para estimar los parámetros de la gramática.
Diferentes alternativas se han propuesto para mejorar la eficiencia del proceso de estimación de los parámetros, la mayoría de ellas basadas en el algoritmo *Inside-Outside*. Infortunadamente el costo computacional de estimación de los parámetros sigue siendo muy elevado.
En este trabajo se presentan los conceptos básicos de los modelos de lenguaje, gramáticas y modelos basados en gramáticas. Se analizan las dificultades computacionales en los algoritmos de estimación de los parámetros que actualmente se usan para los modelos de lenguaje basados en gramáticas libres de contexto estocásticas.

**Palabras y frases claves:** Reconocimiento de formas, teoría de autómatas, modelos de lenguaje, gramáticas formales, gramáticas probabilísticas, funciones de probabilidad

## 1. Introducción

Un modelo estocástico de lenguaje (MLE), es un modelo matemático en el cual se ha definido una función de probabilidad que calcula la probabilidad de ocurrencia de una frase $s$ en un lenguaje dado[(1)]. Si $\Sigma$

[(1)] En las aplicaciones en las que se involucran elementos de lenguaje natural, los elementos del vocabulario se denominan palabras y una cadena finita de elementos del vocabulario se denomina frase.

es un conjunto finito de símbolos (vocabulario), un conjunto de cadenas de longitud finita $w_1 w_2 \dots w_n$ formadas con elementos de $\Sigma$, se denomina un lenguaje. Las cadenas (frases) del lenguaje se denotan $s, x, w$, etc. La expresión $p(s)$ define la probabilidad de ocurrencia de $s$. Más adelante se define formalmente el concepto de lenguaje.

Los parámetros del MLE (probabilidades de las cadenas) son aprendidos a partir de datos. Una cantidad grande de datos en forma de texto, denominada corpus [14], se recolecta y después se utiliza para aprender los parámetro del modelo ( la distribución de probabilidades) automáticamente. Usualmente el corpus se divide en dos partes: el corpus de entrenamiento y el corpus de prueba. El primero es usado para aprender los parámetros y el segundo, para probar la capacidad expresiva del modelo aprendido[(2)].

Uno de los mecanismos para generar Modelos de lenguaje son las gramáticas, en especial las gramáticas libres de contexto (GI) y su extensión estocástica, las gramáticas libres de contexto probabilísticas (GIP). Los ML basados en gramáticas capturan información que no es posible capturar con otros modelos comúnmente usados. Infortunadamente el costo temporal de la estimación de sus parámetros en tareas complejas es elevado.

En este artículo se presentan los conceptos básicos que definen los modelos de lenguaje, se da la definición de gramática formal y gramática libre de contexto, así como gramática libre de contexto probabilística. Se presenta aquí el algoritmo Inside-Outside (IO), que es el método más utilizado en la actualidad para la estimación de los parámetros de una GIP. Se analiza la complejidad computacional del IO y se sugieren alternativas de estimación, que están siendo investigadas en la actualidad.

## 2. Modelos condicionales

Los MLE tradicionales calculan la probabilidad de una frase (cadena) $s = w_1 \cdots w_n$ mediante el uso del teorema de probabilidad total:

$$p(s) = p(w_1 w_2 \dots w_n) = \prod_{i=1}^{n} p(w_i | h_i), \tag{1}$$

donde $h_i$ se denomina la historia de la palabra $w_i$ y esta definida como $h_i = w_1 \dots w_{i-1}$ [4]. El esfuerzo en las técnicas de modelado de lenguaje

[(2)] La capacidad expresiva del modelo es medida entre otras formas, mediante una función de la entropía.

se centra generalmente en el cálculo de $p(w_i|h_i)$ [12]. Este tipo de modelos del lenguaje se denominan modelos condicionales.

Puede observarse que en general la tarea de estimar las probabilidades en (1) es bastante costosa debido al tamaño del vocabulario y al gran número de posibles frases que pueden aparecer. Los MLE tradicionales asumen que la probabilidad de una palabra $w_i$ no depende de la historia completa, y ésta es limitada por una relación de equivalencia $\phi$, así (1) toma la forma:

$$p(s) = p(w_1 w_2 \dots w_n) \approx \prod_{i=1}^{n} p(w_i|\Phi(h_i)), \tag{2}$$

donde $\Phi$ es la clase de equivalencia correspondiente a la historia $h_i$. De esa manera se puede reducir el número de parámetros a estimar en el modelo.

### 2.1. Modelos de $n$-gramas

El modelo condicional que más se ha utilizado es el de *n-gramas* que fué propuesto en 1971 por Bahl y otros [5]. Sorprendentemente, pese a sus limitaciones, hasta el momento se ha mantenido como modelo en la mayoría de las aplicaciones. En el modelo de $n$-gramas la historia se reduce (por la relación de equivalencia) a las últimas $n-1$ palabras, dos historias están en la misma clase si coinciden en las últimas $n-1$ palabras. La probabilidad $p(w_i|\Phi(h_i))$ se calcula mediante:

$$p(w_i|w_{i-n+1}, \dots, w_{i-1}) \simeq \frac{C(w_{i-n+1}, \dots, w_{i-1}, w_i)}{C(w_{i-n+1}, \dots, w_{i-1})}, \tag{3}$$

donde $C(w_{a_1}, \dots, w_{a_k})$ es el número de veces que la cadena $w_{a_1}, \dots, w_{a_k}$ se ha visto en el corpus de entrenamiento.

Entre las características que han hecho del modelo de $n$-gramas una poderosa herramienta se encuentran:

- La fuerte relación que existe en el lenguaje natural entre las restricciones locales.
- La consistencia con los datos de entrenamiento.
- La formulación tan sencilla y la facilidad de implementación.

Pero, por otro lado también el modelo tiene algunas deficiencias:

- Para calcular la probabilidad de una palabra solamente se usa información local, la que dan la $n-1$ palabras anteriores.
- El valor de $n$ debe mantenerse relativamente pequeño ($\leq 3$ ) pues para $n > 3$ se presentan problemas computacionales en la estimación de los parámetros.
- El modelo de $n$-gramas tiene problemas de dispersión: hay gran cantidad de eventos elementales en el espacio de eventos, que no son vistos durante el proceso de estimación y por lo tanto su frecuencia relativa es cero, lo que causaría que a frases con alta probabilidad que contengan dicho evento se les asignara probabilidad cero.
- El modelo de $n$-gramas no se acomoda a los cambios del discurso.

Si se quiere hacer uso de la información no local contenida en la frase, el modelo de $n$-gramas no es el apropiado. Por ejemplo, la información sobre la estructura gramatical contenida en una frase no se puede capturar con el modelo de $n$-gramas. Para solucionar tal deficiencia se han propuesto diferentes tipos de modelos híbridos, en los que se combina la eficiencia local del modelo de $n$-gramas con otros modelos que capturan la información de larga distancia [(3)] [10], usualmente utilizando interpolación lineal.

### 2.2. Modelos de máxima entropía (MLME)

Un marco formal eficiente para incluir información de larga distancia en un único modelo de lenguaje es el denominado principio de máxima entropía (PME). Utilizando el PME se puede combinar información proveniente de muchas fuentes en un mismo modelo de lenguaje [16].

Formalmente el PME puede plantearse en los términos:

Sea $X$ una variable aleatoria que toma valores en un conjunto finito $\mathcal{X}$. Se define:

$$\mathcal{L} = \left\{ p : \sum_{x \in \mathcal{X}} p(x) f_i(x) = K_i \quad i = 1, \ldots, m \right\}, \tag{4}$$

donde $p$ es una distribución de probabilidades sobre $\mathcal{X}$, las $f_i$ son funciones dadas y las $K_i$ son constantes. $\mathcal{L}$ es una familia lineal de distribuciones de probabilidad sobre $\mathcal{X}$. El problema consiste en encontrar

[(3)] La información contenida en la cadena que no puede extraerse de los eventos del modelo de $n$-gramas, se denomina información de larga distancia.

la distribución de probabilidades $\hat{p}$ de $\mathcal{L}$ que haga máxima la entropía $H(p)$. En otros términos, $\hat{p} \in \mathcal{L}$ y

$$H(\hat{p}) = \max_{p \in \mathcal{L}} H(p). \tag{5}$$

La entropía de una distribución de probabilidad $p$ se define como:

$$H(p) = -\sum_{x \in \mathcal{X}} p(x) \log p(x).$$

El modelo de lenguaje de máxima entropía MLME, puede interpretarse en forma intuitiva de la siguiente manera: dado un conjunto de características (piezas de información contenidas en la frase), un conjunto de funciones $f_1, \ldots f_m$ (que miden la contribución de cada característica al modelo) y un conjunto de restricciones[4], debemos encontrar, entre todas las distribuciones de probabilidad para las que se cumplen las restricciones, aquella que haga máxima la entropía.
Resolviendo el problema utilizando multiplicadores de Lagrange se obtiene la solución

$$\hat{p}(x) = p_0(x) \exp\left\{\sum_{i=1}^{n} \lambda_i f_i(x)\right\}, \tag{6}$$

$p_0$ es una distribución de probabilidades *a priori* seleccionada inicialmente. La versión condicional del MLME tiene la forma

$$\hat{p}(y \mid x) = \frac{1}{Z(x)} \exp\left\{\sum_{i=1}^{n} \lambda_i f_i(x, y)\right\}, \tag{7}$$

donde $y$ es una palabra, $x$ es un contexto o historia, $Z(x)$ se denomina constante de normalización (que depende de cada contexto), los $\lambda_i$ son los multiplicadores de Lagrange que ponderan los aportes de cada pieza de información y son estimados mediante un algoritmo iterativo [11].

Los modelos condicionales tienen ciertas limitaciones

- Hay características que representan información contenidas en una frase que no se pueden modelar con los modelos condicionales, por ejemplo su longitud, la coherencia semántica, o las construcciones sintácticas. Características que pueden aportar valiosa información al modelo.

[4] Las restricciones usualmente involucran la igualdad entre el valor esperado teórico y el empírico de cada función $f_i$ calculado en el corpus de entrenamiento.

- Los modelos condicionales asumen independencia al momento de calcular la probabilidad de una frase, es decir: en los modelos condicionales se aproxima $p(w_i|w_1,\ldots,w_{i-1})$ mediante $p(w_i|\Phi(h_i))$ y se asume que la ocurrencia de $w_i$ es independiente de la ocurrencia de algunos de los $w_k$ anteriores. Ésto puede introducir algún error en el cálculo de la probabilidad de $w_i$.

Las desventajas de los modelos condicionales han motivado el estudio de los modelos de frase completa MLFC, en los cuales no se utiliza la regla de Bayes sino que se busca calcular la probabilidad de toda la cadena $s = w_1 \cdots w_n$. Este punto de vista permite incorporar información al modelo que no puede ser incorporada en los modelos condicionales [17, 1]. El más importante de ellos es el modelo de máxima entropía de frase completa MLMEFC que define la ecuación (6)

$$\hat{p}(s) = \frac{1}{Z} p_0(s) \exp\left\{\sum_{i=1}^{n} \lambda_i f_i(s)\right\}, \tag{8}$$

donde $p_0$ es una distribución de probabilidades a priori, $s = w_1 \cdots w_n$, las funciones $f_i$ miden la ocurrencia de la característica $i$-ésima, $Z$ es la constante de normalización y $\lambda_i$ son los parámetros del modelo, que representan la contribución de cada una de las característica al modelo. La estimación de los parámetros se realiza mediante el uso de simulación por cadenas de Markov [2]

## 3. Gramáticas libres de contexto estocásticas

A pesar de que los MLFC tienen mayor capacidad expresiva que los modelos tradicionales [2], no han adquirido gran popularidad dado que su implementación requiere la modificación de otros componentes del sistema que están acoplados a modelos condicionales. Además el costo temporal de su entrenamiento es también elevado. Una alternativa son los modelos basados en gramáticas, que se estudia a continuación.

**Definición 1.** *Una gramática (formal) es una* 4*-upla* $G = (N, \Sigma, P, S)$ *donde:*

- $\Sigma$ *es un conjunto finito, denominado conjunto de terminales (vocabulario).*
- $N$ *es un conjunto finito denominado no terminales,* $\Sigma \cap N = \emptyset$

- *$P$ es un conjunto finito de reglas, llamado conjunto de reglas de producción.*
- *$S$ es el símbolo inicial de la gramática, $S \in N$*

**Definición 2.** *Dado un conjunto $\Sigma$, una cadena de elementos de $\Sigma$ es una secuencia $x_1 \cdots x_n$, $x_i \in \Sigma$.*

**Definición 3.** *La longitud de una cadena $x$ denotada $|x|$ es el número de elementos que tiene la cadena. La cadena vacía es la cadena sin elementos, $|x| = 0$ y se denota $\epsilon$.*

**Definición 4.** *El conjunto de cadenas de $\Sigma$ de longitud mayor o igual a cero se denota $\Sigma^*$. El conjunto de cadenas de $\Sigma$ de longitud mayor o igual a uno se denota $\Sigma^+$.*

**Definición 5.** *Un lenguaje $L$ es un subconjunto de $\Sigma^*$.*

**Ejemplo 1.** *Sea $G = (N, \Sigma, P, S)$, donde: $N = \{A, B, C, D, E, S\}$, $\Sigma = \{a\}$, y las reglas de producción $P$ consisten en:*

| | | | |
|---|---|---|---|
| 1 | $S \to ACaB$ | 5 | $aD \to Da$ |
| 2 | $Ca \to aaC$ | 6 | $AD \to AC$ |
| 3 | $CB \to DB$ | 7 | $aE \to Ea$ |
| 4 | $CB \to E$ | 8 | $AE \to \epsilon$ |

Una derivación a partir de una cadena es la aplicación de una regla de $P$ a la cadena. Por ejemplo a partir de la cadena $S$ aplicando la regla 1 se obtiene la cadena $ACaB$ y lo notamos $S \Rightarrow ACaB$, si a la última cadena le aplicamos la regla 2 obtenemos $ACaB \Rightarrow AaaCB$.

**Definición 6.** *Una gramática $G$, para la cual el conjunto $P$ de reglas de derivación esta constituido por reglas de la forma $A \to \alpha$ donde $A \in N$ y $\alpha \in (N \cup \Sigma)^+$, se denomina gramática libre de contexto (GI).*

Solamente se considerarán gramáticas en forma normal de Chomsky, es decir: gramáticas en las que las reglas tienen la forma $A \to BC$ o $A \to v$ donde $A, B, C \in N$ y $v \in \Sigma$.

Una derivación a izquierda de una cadena $x \in \Sigma^+$ mediante $G$, es una sucesión de reglas de producción $d_x = (q_1, q_2, \ldots, q_m)$ con $m \geq 1$, tal que: $\left(S \overset{q_1}{\Rightarrow} \alpha_1 \overset{q_2}{\Rightarrow} \alpha_2, \ldots \overset{q_m}{\Rightarrow} x\right)$, donde $\alpha_i \in (N \cup \Sigma)^+$, $1 \leq i \leq m-1$ y $q_i$ reescribe el terminal más a la izquierda de $\alpha_{i-1}$.

**Definición 7.** *El lenguaje generado por $G$ es el conjunto $L(G)$, definido como: $L(G) = \left\{x \in \Sigma^+ | S \overset{*}{\Rightarrow} x\right\}$. Donde $S \overset{*}{\Rightarrow} x$ significa que $x$ se ha obtenido, o derivado, a partir de $S$ mediante derivaciones a izquierda por medio de la gramática $G$.*

Es oportuno aclarar que una cadena puede ser obtenida por diferentes derivaciones.

### 3.3. Gramáticas libres de contexto probabilísticas

Una gramática libre de contexto probabilística (GIP) es un par $(G, p)$ donde $G$ es una gramática libre de contexto y $p$ es una función definida sobre las reglas, $p : P \longrightarrow (0, 1]$, tal que

$$\forall A \in N \qquad \sum_{(A \to \alpha) \in \Gamma_A} p(A \to \alpha) = 1,$$

donde $\Gamma_A$ representa el conjunto de reglas de la gramática con antecedente $A$. La GIP suele denotarse como $G_p = (G, p)$

**Ejemplo 2.**

| | | | |
|---|---|---|---|
| $S \to AC$ | 0,4 | $C \to SA$ | 1,0 |
| $S \to BD$ | 0,4 | $D \to SB$ | 1,0 |
| $S \to AA$ | 0,1 | $A \to a$ | 1,0 |
| $S \to BB$ | 0,1 | $B \to b$ | 1,0 |

Las GIPs representan un mecanismo eficiente para modelar las relaciones de larga distancia entre las diferentes unidades léxicas en una frase. Las GIPs han sido utilizadas en diferentes tareas dentro de campos como el reconocimiento sintáctico de formas y lingüística computacional. Las GIPs han sido utilizadas con éxito en tareas limitadas, sin embargo, las GIPs de propósito general no se comportan adecuadamente en tareas complejas con grandes vocabularios.

Uno de los principales obstáculos para el uso de las GIPs en tareas complejas radica en el proceso de aprendizaje del modelo. Dos aspectos hay que considerar al respecto: el aprendizaje del componente estructural, es decir las reglas de la gramática y la estimación del componente estocástico, es decir la probabilidad de las reglas.

Existen técnicas robustas que permiten estimar de manera automática las probabilidades de las reglas de las GIPs. Una de las más conocidos hace uso del teorema de transformaciones crecientes junto con el algoritmo *Inside-Outside* (IO) [9]. Desafortunadamente, existen serias limitaciones para su aplicación: la complejidad temporal por iteración y el alto número

de iteraciones necesarias para obtener la convergencia. Una alternativa al algoritmo IO es un algoritmo basado en el algoritmo *Viterbi Score* (VS) [19]. La convergencia del algoritmo VS es más rápida que la del algoritmo IO, pero en general, la gramática resultante no es bien aprendida.

Se han propuesto otras alternativas para estimar los parámetros de la GIP. En dichas alternativas se considera solamente un subconjunto de las derivaciones en el proceso de estimación. Para seleccionar tal subconjunto, se han considerado dos aproximaciones.

- Estructural, a partir de la información contenida en corpus etiquetados [3, 15]. En esta aproximación se modifican el algoritmos IO para que en el proceso de entrenamiento se tengan en cuenta solamente las cadenas compatibles con el etiquetado.
- Estadística, mediante la información contenida en las $k$-mejores derivaciones [18].

### 3.4. Probabilidad de una cadena

Dada una derivación de una cadena, la probabilidad de tal derivación se define como

$$\Pr(x, d_x|G_p) = \prod_{(A\to\alpha)\in P} p(A \to \alpha)^{N(A\to\alpha, d_x)}, \tag{9}$$

donde $N(A \to \alpha, d_x)$ es el número de veces que la regla $A \to \alpha$ se ha utilizado en la derivación.
La probabilidad de una cadena se define como

$$\Pr(x|G_p) = \sum_{d_x\in D_x} \Pr(x, d_x|G_p), \tag{10}$$

donde $D_x$ es el conjunto de todas las derivaciones posibles de la cadena. El lenguaje generado por una gramática es definido por

$$L(G_p) = \{x \in L(G) : \Pr(x|G_p) > 0\}.$$

### 3.5. Estimación de los parámetros de una GIP

Para poder definir un modelo de lenguaje basado en una GIP hay primero estimar el valor de las probabilidades de sus reglas, es decir estimar los parámetros. El problema de la estimación de los parámetros de una GIP $G_p = (G, p)$ puede plantearse de la siguiente manera: sea $(L, \Phi)$

un lenguaje estocástico donde $L \subseteq L(G)$, y $\Phi$ es una función de probabilidad desconocida tal que $\sum_{x \in L} \Phi(x) = 1$. Dada una muestra $\Omega$ de $L$, se tiene que estimar $p$ (la probabilidad de las reglas) de tal manera que represente a $\Phi$. Si se asume que $\Phi$ es representada mediante la gramática $G_p$, se quiere obtener:

$$\hat{p} = \arg \max_{p} f_p(\Omega),$$

donde $f_p$ es una función a ser optimizada. De tal manera que para poder estimar los parámetros de una GIP, es necesario definir la función a optimizar y el método de optimización que se ha de utilizar.

Respecto al método de optimización tradicionalmente se considerara el método de transformaciones crecientes [6], basado en el siguiente teorema:

**Teorema 1.** *Sea $P(\Theta)$, $\Theta = \{\Theta_{ij}\}$, un polinomio homogéneo de grado $d$ con coeficientes no negativos. Sea $\theta = \{\theta_{ij}\}$ un punto del conjunto $D = \left\{\theta_{ij} | \theta_{ij} \geq 0, \sum_{j=1}^{q_i} \theta_{ij} = 1, i = 1, \ldots, k \quad j = 1, \ldots, q_i\right\}$ para $k$, $q_i$ enteros. Sea $Q(\Theta)$ un punto del conjunto $D$ definido como*

$$Q(\theta)_{ij} = \frac{\theta_{ij} \left(\frac{\partial P}{\partial \Theta_{ij}}\right)_{\theta}}{\sum_{k=1}^{q_i} \theta_{ik} \left(\frac{\partial P}{\partial \Theta_{ik}}\right)_{\theta}},$$

*tal que para todo $i$, $\sum_{k=1}^{q_i} \theta_{ik} \left(\frac{\partial P}{\partial \Theta_{ik}}\right)_{\theta} \neq 0$. Entonces, $P(Q(\theta)) > P(\theta)$ para $Q(\theta) \neq \theta$.*

En el caso que nos ocupa, la función a optimizar es la verosimilitud de $\Omega$, definida como

$$L(\Omega | G_p) = \prod_{x \in \Omega} \Pr(x | G_p). \tag{11}$$

Como caso particular, las funciones de probabilidad sobre las reglas que definen la GIP, pertenecen al conjunto $D$ descrito en el teorema 1 y la función de verosimilitud es un polinomio multivariado, entonces ella puede ser optimizada aplicando iterativamente el teorema 1.
La aplicación del teorema 1 al caso de la función de verosimilitud nos conduce a las siguientes ecuaciones

$$\hat{p}(A \to \alpha) = \frac{\Pr(A \to \alpha) \left(\frac{\partial L(\Omega | G_p)}{\partial \Pr(A \to \alpha)}\right)_p}{\sum_{(A \to \alpha) \in \Gamma_A} \Pr(A \to \alpha) \left(\frac{\partial L(\Omega | G_p)}{\partial \Pr(A \to \alpha)}\right)_p}. \tag{12}$$

Resolviendo las derivadas en la expresión anterior, utilizando (9) y (12) y luego de reducciones algebraicas, se tiene

$$\hat{p}(A \to \alpha) = \frac{\sum_{x\in\Omega} \frac{1}{\Pr(x|G_p)} \sum_{d_x \in D_x} N(A \to \alpha, d_x) \Pr(x, d_x|G_p)}{\sum_{x\in\Omega} \frac{1}{\Pr(x|G_p)} \sum_{d_x \in D_x} N(A, d_x) \Pr(x, d_x|G_p)}, \quad (13)$$

donde $N(A, d_x) = \sum_{(A\to\alpha)\in\Gamma_A} N(A \to \alpha, d_x)$ es el número de veces que el no terminal $A$ ha sido derivado en $d_x$.

### 3.6. Algoritmo Inside-outside

Los algoritmos Inside (I), Outside (O) e Inside-Outside (IO) permiten calcular recursivamente las probabilidades $\Pr(x, d_x|G_p)$, que aparecen en (13), mediante el siguiente esquema[5]:

**Algoritmo Inside**

La expresión

$$e(A < i, j >) = \Pr(A \overset{*}{\Rightarrow} w_i \dots w_j | G_p)$$

define la probabilidad de que la subcadena $w_i \dots w_j$ sea generada a partir de $A$ dada la gramática $G_s$, y se define recursivamente como

$$e\left(A\left\langle i,i\right\rangle\right) = p\left(A \to x_i\right)$$

$$e\left(A\left\langle i,j\right\rangle\right) = \sum_{B,C\in N} p\left(A \to BC\right) \sum_{k=i}^{j-1} e\left(B\left\langle i,k\right\rangle\right) e\left(C\left\langle k+1,j\right\rangle\right), \qquad 1 \le i \le j \le |x|$$

de ésta forma, $Pr\left(x|G_p\right) = e\left(S\left\langle 1,|x|\right\rangle\right)$.

**Algoritmo Outside**

La expresión $f(A < i, j >) = \Pr(S \overset{*}{\Rightarrow} x_1 \dots x_{i-1} A x_{i+1} \dots x_n | G_p)$ es la probabilidad de que a partir del símbolo inicial se genere la cadena $x_1 \dots x_{i-1}$ a continuación el no terminal $A$ y a continuación la cadena $x_{i+1} \dots x_n$ y se calcula mediante el esquema

$\forall A \in N$

[5] Supondremos que la gramática esta en FNC.

$$f(A<1,|\ x\ |>)=\begin{cases}1, & si \quad A=S\\ 0, & si \quad A\neq S\end{cases}$$

$$f(A<i,j>)=\sum_{B,C\in N}(p(B\longrightarrow CA)\sum_{k=1}^{i-1}f(B<k,j>)e(C<k,i-1>)$$
$$+p(B\longrightarrow AC)\sum_{k=j+1}^{|x|}f(B<i,k>)e(C<j+1,k>))$$
$$1\leq i\leq j\leq |x|$$

De esta forma, $Pr\,(x|G_p)=\sum_{A\in N}f\,(A\,\langle i,i\rangle)\,p(A\to x_i),\quad 1\leq i\leq |x|$.

**Algoritmo Inside-Outside (IO)**

Si la gramática bajo estudio esta en FNC, consideremos una regla de la forma $A\to BC$ con $A,B,C\in N$. Si $d_x$ es una derivación de la cadena $x$, y $t_x$ el árbol de derivación correspondiente, de manera que la regla $A\to BC$ aparece en $t_x$ en una posición delimitada por los enteros $i,j,k$, $1\leq i\leq k\leq j\leq |x|$ y definiendo $g(x)=\frac{p(A\to BC)}{\Pr(x|G_p)}$ tenemos que

$$\hat{p}(A\to BC)=\frac{\sum\limits_{x\in\Omega}g(x)\sum\limits_{1\leq i\leq k\leq j\leq |x|}f(A\langle i,j\rangle)e(B\langle i,k\rangle)e(C\langle k+1,j\rangle)}{\sum\limits_{x\in\Omega}\frac{1}{\Pr(x|G_p|)}\sum\limits_{i=1}^{|x|}\sum\limits_{j=i}^{|x|}f(A\langle i,j\rangle)e(A\langle i,j\rangle)} \tag{14}$$

y para las reglas de la forma $A\to a\in P$

$$\hat{p}(A\to a)=\frac{\sum_{x\in\Omega}\frac{1}{\Pr(x|G_p)}\sum_{i=1,a=x_i}f(A\langle i,i\rangle)p(A\to x_i)}{\sum_{x\in\Omega}\frac{1}{\Pr(x|G_p|)}\sum_{i=1}^{|x|}\sum_{j=i}^{|x|}f(A\langle i,j\rangle)e(A\langle i,j\rangle)}. \tag{15}$$

La aplicación reiterada del teorema 1 a la función (11), comenzando con una asignación inicial de probabilidades a las reglas de la gramática, que en este caso serían las variables del polinomio, nos conduce a un punto en el que la verosimilitud es máxima, Es decir a la mejor asignación de probabilidades a las reglas.

El costo temporal del algoritmo IO es $O(\mid \Omega \mid l_m^3 \mid P \mid)$ donde $l_m$ es la longitud de la cadena más grande [9]. En el peor de los casos la cantidad de reglas $\mid P \mid \in O(\mid N \mid^3)$. Su costo espacial es $O(l_m^2 \mid N \mid)$.
Para vocabularios de tamaño relativamente grande el costo temporal es bastante elevado. En experimentos con lenguaje natural, inglés, se han utilizado vocabularios de 25,000 palabras, corpus extraído del *Wall Street Journal*, proyecto *Penn Treebanck* [14], el costo temporal en la estimación de los parámetros es tan elevado que hace poco práctica la utilización de tal modelo.

## 4. Conclusiones

A partir de la complejidad temporal del algoritmo IO puede observarse que si en una aplicación el tamaño del vocabulario es grande y en consecuencia el número de reglas de la gramática inicial también lo es, el proceso de entrenamiento es bastante costoso; días o meses, lo cual hace que no sea factible utilizar una gramática como vehículo para un modelo de lenguaje, a pesar de sus buenas propiedades. Aún cuando los sistemas de cómputo evolucionan en forma acelerada, todavía no se esta en condiciones de, por ejemplo, entrenar modelos de lenguaje basados en gramáticas *on-line*.
En los últimos años se han realizado progresos en la reducción del costo temporal de entrenamiento mediante la utilización de técnicas como por ejemplo, la agrupación del vocabulario en clases , o utilizando las $k$ mejores derivaciones de una palabra en lugar de todas las derivaciones. Pero aun persisten problemas computacionales.
Actualmente los autores están trabajando en una solución al problema de la estimación cambiando el paradigma de optimización de transformaciones crecientes, por otros métodos de optimización de polinomios: por un lado reduciendo el problema a un problema de optimización convexa LMI (Linear Matrix inequality) mediante el método de los momentos [13] y por otro mediante métodos cuasi-Newton para problemas con muchas variable o restricciones [7].

## Referencias

[1] F. Amaya y J. M. Benedí. Using Perfect Sampling in Parameter Estimation of a Wole Sentence Maximum Entropy Language Model. *Proc. Fourth Computational Natural Language Learning Workshop, CoNLL-2000, Lisbon, Portugal*, 2000.

[2] F. Amaya y J. M. Benedí. Improvement of a Whole Sentence Maximumn Entropy Language Model using Grammatical Features. *Proc. of the 39th meeting of the Association for Computational Linguistics (ACL-2001), Toulouse, France*, 2001.

[3] F. Amaya, J. A. Sanchez, y J. M. Benedí. Learning stochastic context-free grammars from bracketed corpora by means of reestimation algorithms. *Proc. VIII Spanish Symposium on Pattern Recognition and Image Analysis, Bilbao, Spain*, pages 119–126, 1999.

[4] L.R. Bahal, F.Jelinek, y R. L. Mercer. A maximun likelihood approach to continuous speech recognition. *IEEE Trans. on Pattern analysis and Machine Intelligence*, 5(2):179–190, 1983.

[5] L. R. Bahl, J. K. Baker, P. S. Cohen, F. Jelinek, B. L. Lews, y R.L. Mercer. Recognition of a continuously read natural corpus. *Proc. IEEE Int. Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1978.

[6] L. E. Baum. An inequality and associated maximization technique in statistical estimation for probabilistic functions of markov processes. *Inequalities*, 3:1–8, 1972.

[7] L.T. Biegler, J. Nocedal y C. Schmid. A reduced Hessian method for large-scale constrained optimization. *SIAM Journal of Optimization)*, 1993.

[8] P.F. Brown, J. Cocke, R.L. Mercer, V.J. Della Pietra, S. Della Pietra, F. Jelinek, J. D. Lafferty, y P. Rossinn. A statistical approach to machine translation. *Computational Linguistics*, 16(2), 1990.

[9] F. Casacuberta. Statistical estimation of stochastic context-free grammars. *Pattern Recognition Letters*, 16:565–573, 1995.

[10] C. Chelba y F. Jelinek. Exploting syntactic structure for language model. *COLING-ACL*, 1998.

[11] S. Della Pietra, V. Della Pietra, y J. Lafferty. Inducing features of random fields. Technical Report CMU-CS-95-144, Carnegie Mellon University, 1995.

[12] F. Jelinek. *Statistical Methods for Speech Recognition*. The MIT Press, Massachusetts Institut of Technology. Cambridge, Massachusetts, 1997.

[13] J. B. Lasserre. Global optimization with polynomials and the problem of moments. SIAM J. OPTIM., pages 786–817. Vol 11, 2001.

[14] M. P. Marcus, B. Santorini, y M.A. Marcinkiewicz. Building a large annotated corpus of English: the Penn Treebanck. *Computational Linguistics*, 19, 1993.

[15] F. Pereira y Y. Shabes. Inside-outsude reestimation from partially bracketed corpora. *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*, pages 128–135, 1992. University of Delaware.

[16] R. Rosenfeld. A Maximun Entropy approach to adaptive statistical language modeling. *Computer Speech and Language*, 10:187–228, 1996.

[17] R. Rosenfeld. Incorporing linguistic structure into statistical language models. *Philosofical transactions of the Royal Society, Series A*, pages 1311–1324, 2000.

[18] J. A. Sánchez y J. M. Benedí. Estimation of the probability distribution of stochastic context-free grammars from the k-best derivations. *5th International Conference on Spoken Language Processing*, pages 2495–2498, 1998.

[19] J.A. Sánchez, J.M. Benedí, y F. Casacuberta. Comparison between the inside-outside algorithm and the viterbi algorithm for stochastic context-free grammars. In P. Perner, P. Wang, y A. Rosenfeld, editors, *Advances in Structural and Syntactical Pattern Recognition*, pages 50–59. Srpringer-Verlag, 1996.

*Dirección del autor:* Fredy Amaya Robayo Departamento de Matemáticas Universidad Del Cauca Popayán, Colombia famaya@ucauca.edu.co