Channel Capacity of Equal Matrix Languages

RANI SIROMONEY

Madras Christian College, Madras 59, India

The channel capacity of the class of equal matrix languages which includes several well-known context-sensitive and context-free languages is found out.

Equal matrix languages (abbreviated EML) have the interesting property that the corresponding Parikh mappings are semilinear (Siromoney, 1969a, 1969b). Further, every equal matrix grammar (abbreviated EMG) is nonterminal bounded in the sense that in any line of a derivation the total number of nonterminals is less than a constant. This concept of nonterminal boundedness was first introduced by Banerji (1963) to study the channel capacity of a certain class of context-free languages (abbreviated CFL). We extend the calculation of channel capacity to the class of unambiguous, uniquely deconcatenable EML’s, which includes several of the well-known context-sensitive languages.

Let \( \Sigma \) be a finite nonempty set and \( \Sigma^* \) the free semigroup with identity \( e \) generated by \( \Sigma \). An equal matrix grammar is a 4-tuple \( G = (V, \Sigma, P, S) \) where

1. the total vocabulary \( V \) consists of the terminal vocabulary \( \Sigma \), a set \( V_N \) of nonterminals (variables) consisting of the initial symbol \( S \) and a set of distinct \( k \)-tuples \( \langle A_1, \cdots, A_k \rangle \).

2. \( P \) consists of the following types of matrix rules:
   (i) a set of initial matrix rules of the form \([S \rightarrow e]\) or \([S \rightarrow f_1 A_1 \cdots f_k A_k]\), where \( f_1, \cdots, f_k \) (possibly empty) are elements of \( \Sigma^* \) and \( \langle A_1, \cdots, A_k \rangle \) is in \( V_N \).
   (ii) a set of nonterminal matrix rules of the form
   
   \[
   \begin{bmatrix}
   A_1 & \rightarrow & f_1 B_1 \\
   \cdots & \cdots & \cdots \\
   A_k & \rightarrow & f_k B_k 
   \end{bmatrix}
   \]

where \( f_1, \cdots, f_k \in \Sigma^* \) are not all empty and \( \langle A_1, \cdots, A_k \rangle, \langle B_1, \cdots, B_k \rangle \) are in \( V_N \).
(iii) a set of terminal matrix rules of the form

\[
\begin{align*}
A_1 & \rightarrow f_1 \\
\vdots & \\
A_k & \rightarrow f_k
\end{align*}
\]

\(f_1, \ldots, f_k\) and \(\langle A_1, \ldots, A_k \rangle\) as in (ii).

The EMG is said to be of order \(k\).

A matrix rule \(p\) is applicable to a string \(\phi\) if \(\phi\) contains in the given order every element of the \(k\)-tuple which occurs on the left side of \(p\) and the result is obtained by replacing every element on the left side by the corresponding string on the right side. Thus the rule

\[
\begin{align*}
A_k & \rightarrow f_k B_k \\
\vdots & \\
A_1 & \rightarrow f_1 B_1
\end{align*}
\]

can be applied to the string \(x_1 A_1 \cdots x_k A_k\) and the resultant is \(x_1 f_1 B_1 \cdots x_k f_k B_k, x_i, f_i \in \Sigma^*, \langle A_1, \ldots, A_k \rangle, \langle B_1, \ldots, B_k \rangle \in V_N\).

In a grammar, if \(\psi\) is obtained as the resultant of \(\phi\), we write \(\phi \Rightarrow \psi\), and if there is a sequence of strings \(\phi_0, \phi_1, \ldots, \phi_n\) with \(\phi_i \Rightarrow \phi_{i+1}\) for each \(i\), we write \(\phi_0 \Rightarrow^* \phi_n\). \(\phi_0 \Rightarrow \phi_1 \Rightarrow \cdots \Rightarrow \phi_n\) is called the \(\phi_0\) derivation or generation of \(\phi_n\), and \(\phi_0\) is said to generate \(\phi_n\). Each \(\phi_i\) is called a step (or line) of the derivation. In any grammar \(G\), the set of terminal words generated by \(S\) is called the language generated by \(G\). Thus \(L\) is an EML if there is an EMG \(G\) such that \(L = \{w \in \Sigma^* \mid S \Rightarrow^* w\text{ is a derivation in } G\}\) and \(L\) is CF if the grammar generating \(L\) is CF.

To illustrate the idea of an EML we consider the context-sensitive language \(L_3 = \{\text{a string of } a\text{'s and } b\text{'s followed by an equal string}\}\) (Chomsky, 1963). This is generated by the EMG consisting of the rules,

\[
[S \rightarrow AB], \quad \begin{bmatrix} A \rightarrow aA \\ B \rightarrow aB \end{bmatrix}, \quad \begin{bmatrix} A \rightarrow a \\ B \rightarrow a \end{bmatrix}, \quad \begin{bmatrix} A \rightarrow bA \\ B \rightarrow bB \end{bmatrix}, \quad \begin{bmatrix} A \rightarrow b \\ B \rightarrow b \end{bmatrix}.
\]

The string \(aabaab\) in \(L_3\) is generated by the following derivations:

\(S \Rightarrow AB \Rightarrow aAaB \Rightarrow aaAaaB \Rightarrow aabaab\). i.e. \(S \Rightarrow^* aabaab\).

A CFG is nonterminal bounded if there is an integer \(K\) such that for all \(S \Rightarrow^* \phi, \phi \in V^*\), then the number of nonterminals in \(\phi\) is less than or equal to \(K\). This definition was originally given for CFG but it is clear that it can be extended to grammars that are not CF, including matrix grammars. It is easily seen that an EMG is nonterminal bounded. Banerji (1963) has given a method of calculating the channel capacity of
unambiguous, uniquely deconcatenable, nonterminal bounded CFL and we follow an analogous method of associating with every EMG a finite state automaton (Rabin and Scott, 1959).

Corresponding to every EMG $G = (V, \Sigma, P, S)$ we associate the finite automaton $A = (K, \Sigma', \delta, \Lambda_0, \Lambda_0)$, where $K = V - \Sigma \cup \{\Lambda_0, \Lambda_1\}$ ($\Lambda_0, \Lambda_1$ are new symbols not in $V$), $\Sigma' = \{p_1, \ldots, p_n\}$ where $p_1, \ldots, p_n$ are the matrix rules in $P$, $\Lambda_0$ is the initial and the final state and $\delta$ is defined as follows:

$$\delta(\Lambda_0, \epsilon) = S, \delta(S, p_i) = \langle A_{i1}, \ldots, A_{ki} \rangle$$

corresponding to every initial rule $p_i$ in $P$ of the form $[S \rightarrow e_iA_{i1} \ldots e_iA_{ki}]$, $\delta(\langle A_{i1}, \ldots, A_{ki}, p_i \rangle, p_i) = \langle B_{i1}, \ldots, B_{ki} \rangle$ corresponding to every nonterminal rule $p_i$ in $P$ of the form

$$\begin{bmatrix}
A_{i1} \rightarrow f_{i1}B_{i1} \\
\ldots \ldots \\
A_{ki} \rightarrow f_{ki}B_{ki}
\end{bmatrix},$$

$\delta(\langle A_{i1}, \ldots, A_{ki}, p_i \rangle, p_i) = \Lambda_1$ for every terminal rule $p_i$ in $P$ of the form

$$\begin{bmatrix}
A_{i1} \rightarrow f_{i1} \\
\ldots \ldots \\
A_{ki} \rightarrow f_{ki}
\end{bmatrix}$$

and $\delta(\Lambda_1, \epsilon) = \Lambda_0$.

For every unambiguous, uniquely deconcatenable EML $L$, a finite automaton $A$ can be constructed corresponding to the unambiguous EMG generating $L$. $A$ can be used to calculate the channel capacity of $L$ where the information theoretic definition of the channel capacity $C$ of a language is given by $C = \lim_{t \to \infty} \log N(t)/t$, $N(t)$ being the number of sequences of words of length $t$ in $L$. Hence for the calculation of channel capacity, we need to know the value of $N(t)$. This is calculated as follows from the associated finite automaton. The length $l_T$ of the transformation $\delta(\langle A_{i1}, \ldots, A_{ki}, p_i \rangle, p_i) = \langle B_{i1}, \ldots, B_{ki} \rangle$ is defined to be equal to the length $l_T$ of $p_i$, which is equal to the sum of the number of terminals in $p_i$. For example if $p_i$ is the rule

$$\begin{bmatrix}
A_{i1} \rightarrow f_{i1}B_{1i} \\
\ldots \ldots \\
A_{ki} \rightarrow f_{ki}B_{ki}
\end{bmatrix},$$

then $l_T(p_i) = l(f_{i1}) + \cdots + l(f_{ki})$, where $l(f_{ri})$ is the number of terminals in $f_{ri}$. Length of a word $p_1 \cdots p_n$ accepted by $A$ is equal to $l_T(p_1) + \cdots + l_T(p_n)$. 
If the automaton $A$ has $m$ states $S_1, \ldots, S_m$, $S_1$ being the initial state, let $n_{ij}$ represent the number of transformations from $S_i$ to $S_j$ and let the lengths of these transformations be $l_{ij}(1 \leq i, j \leq m; 1 \leq \alpha \leq n_{ij})$ and $N_n(t)$ the number of strings of length $t$, accepted by $A$. Then $N_n(t) = A_{ij}^{(1)}X_1^t + A_{ij}^{(2)}X_2^t + \cdots + A_{ij}^{(n)}X_n^t$ where the $X_r(1 \leq r \leq n)$ are the solutions of the determinant equation

\[
\begin{vmatrix}
\sum_{\alpha=1}^{n_{11}} X^{-l_{11}^\alpha} & \sum_{\alpha=1}^{n_{12}} X^{-l_{12}^\alpha} & \cdots & \sum_{\alpha=1}^{n_{1m}} X^{-l_{1m}^\alpha} \\
\sum_{\alpha=1}^{n_{21}} X^{-l_{21}^\alpha} & \sum_{\alpha=1}^{n_{22}} X^{-l_{22}^\alpha} & \cdots & \sum_{\alpha=1}^{n_{2m}} X^{-l_{2m}^\alpha} \\
\vdots & \vdots & \ddots & \vdots \\
\sum_{\alpha=1}^{n_{m1}} X^{-l_{m1}^\alpha} & \sum_{\alpha=1}^{n_{m2}} X^{-l_{m2}^\alpha} & \cdots & \sum_{\alpha=1}^{n_{mm}} X^{-l_{mm}^\alpha} - 1
\end{vmatrix} = 0
\]

and the $A_{ij}^{(r)}$ are determined from the boundary values obtained by enumeration for small values of $t$ (Banerji, 1963).

We illustrate the method by calculating the channel capacity of the EML $L = \{a^n b^n c^n \mid n \geq 1\}$ which is known to be context-sensitive (Chomsky, 1963). $L$ is generated by the EMG $G = (V, \Sigma, P, S)$ where $V = \{S, \langle A, B, C \rangle, a, b, c\}$, $\Sigma = \{a, b, c\}$, $S$ the initial symbol and $P$ consists of the rules

\begin{align*}
p_1 &: [S \rightarrow ABC], \\
p_2 &: \begin{bmatrix} A \rightarrow aA \\ B \rightarrow bB \\ C \rightarrow cC \end{bmatrix}, \\
p_3 &: \begin{bmatrix} A \rightarrow a \\ B \rightarrow b \\ C \rightarrow c \end{bmatrix}.
\end{align*}

The associated finite automaton

$A = (K, \Sigma', \delta, \Lambda_0, \Lambda_0)$ where $K = \{\Lambda_0, \Lambda_1, S, \langle A, B, C \rangle\}$,

$\Sigma' = \{p_1, p_2, p_3\}$, $\Lambda_0$ the start and final state. Write $\Lambda_0 = S_1$, $S = S_2$, $\langle A, B, C \rangle = S_3$ and $\Lambda_1 = S_4$. $\delta$ is defined as follows: $\delta(S_1, \epsilon) = S_2$, $\delta(S_2, p_1) = S_3$, $\delta(S_3, p_2) = S_4$, $\delta(S_4, p_3) = S_1$, $\delta(S_4, \epsilon) = S_1$. It is seen that $l_\tau(p_1) = 0$, $l_\tau(p_2) = 3$, $l_\tau(p_3) = 3$, $l_\tau(\epsilon) = 0$. The determinant equation is

\[
\begin{vmatrix}
-1 & 1 & 0 & 0 \\
0 & -1 & 1 & 0 \\
0 & 0 & X^{-3} & 1 \\
1 & 0 & 0 & -1
\end{vmatrix} = 0
\]
i.e. $2X^{-3} - 1 = 0$. Therefore $X = 2^{1/3}, 2^{1/3} \omega, 2^{1/3} \omega^2$ where $\omega$ is the cube root of unity. Hence $N_{11}(t) = 2^{1/3}(A_0 + A_1\omega^t + A_2\omega^{2t})$. Solving for $A_0, A_1, A_2$ from the boundary conditions, $N_{11}(1) = 0 = N_{11}(2), N_{11}(3) = 1$, we get $A_0 = A_1 = A_2 = \frac{1}{6}$, and $N_{11}(t) = (2^{1/3}/6) \cdot (1 + \omega^t + \omega^{2t})$ which is zero when $t$ is not a multiple of 3 and equal to $2^{m-1}$ if $t = 3m$. The channel capacity is $\frac{1}{4}$ bit per symbol.

We have seen that for every EMG, we can associate a finite state automaton, but the method of calculating the channel capacity is restricted to unambiguous, uniquely deconcatenable EML. An EMG $G$ is ambiguous if there is a word $w$ generated by $G$, using more than one derivation. Hence for each word ambiguously derivable in $G$, there will correspond more than one word accepted by the associated finite state automaton. As seen earlier, the context-sensitive language $L_3$ can be generated by an unambiguous EMG but due to lack of unique deconcatenability $N_{11}(t)$ cannot be calculated by this method. Nevertheless, we note that $N_{22}(t)$ can be calculated using the corresponding automaton and it yields the number of strings of length $t$ (instead of the number of sequences of strings of length $t$ which is $N_{11}(t)$).

ACKNOWLEDGMENT

I wish to thank Dr. Gift Siromoney for his help in the preparation of this paper and a referee for his useful suggestions.

Received: August 21, 1968; revised March 18, 1969.


