UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE CIENCIAS

CONCEPTOS DE LENGUAJES BORROSOS Y APLICACIONES

T E S I S

Que para obtener el titulo de:

M A T E M A T I C O

P r e s e n t a :

BERNABE ORTIZ Y HERBERT

México, D. F. 1982

TESIS CON FALLA DE ORIGEN

Conceptos De Lenguajes Borrosos Y Aplicaciones

Conceptos De Lenguajes Borrosos Y Aplicaciones

1.- INTRODUCCION

El propósito principal de escribir esta tesis es la de proporcionar los aspectos fundamentales de los lenguajes borrosos.

El estudio de los lenguajes borrosos se inicia analizando el lenguaje natural y su relacion con respecto a la teoria de lenguajes formales.

La teoría de lenguajes borrosos es desarrollada por primera vez por L. A. Zadeh y es presentada como una manera de tratar una serie de problemas que surgen en lenguajes naturales, como son: La necesidad de elegir un reducido número de sentencias válidas, de la infinidad que se generan a partir de la gramática (sintáxis) que constituye al lenguaje natural, y la frecuencia con que se presentan ambiguedades o imprecisiones en el significado (semántica) de las expresiones en el lenguaje.

Por lo tanto la teoría de lenguajes borrosos tiene por objeto presentar una alternativa que caracterice con precisión los dos aspectos anteriores de los lenguajes naturales a saber: la forma (gramática) y el significado (semántica) donde para el primer aspecto se recurre a la teoría de lenguajes formales y para el segundo a la teoría de conjuntos borrosos.

La teoría de lenguajes formales tiene como principal objetivo, establecer un modelo matemático de las propiedades gramaticales de los lenguajes naturales, y ha permitido el desarrollo de algunos lenguajes de computación, sin embargo tales lenguajes presentan grandes limitaciones como medio expresivo por su incapacidad para la representación del significado, la complementación de esta parte de los lenguajes formales es desarrollada por la teoría de los lenguajes borrosos mediante los conjuntos borrosos.

Conceptos de Lenguajes Borrosos y Aplicaciones

En 1965 L. A. Zadeh desarrolla la teoría de conjuntos borrosos los cuales estan caracterizados por una función que asigna a cada objeto en un conjunto universal, un grado de membresia (característica, pertenencia, etc.) en el intervalo [0,1], de esta manera un conjunto borroso viene a constituir una clase como una sucesión de grados de membresia. Esencialmente la estructura de los conjuntos borrosos proporciona un método de tratar con los problemas en los cuales existe imprecision para establecer la membresia de los objetos en un conjunto. Esta estructura es utilizada por los lenguajes borrosos para precisar el significado de las expresiones en un lenguaje natural.

Por lo tanto los lenguajes borrosos para su estudio constan de una serie de reglas sintácticas y semánticas borrosas, que proporcionan un algoritmo para calcular el grado de membresia de cualquier cadena de caracteres, para determinar el grado gramatical y semántico entre lo correcto e incorrecto.

A lo largo de este trabajo se ha incluido una gran variedad de ejemplos de los conceptos mas importantes, que aparecen.

La primera parte consiste de los conceptos fundamentales de los lenguajes naturales, tomando el español como referencia. Se establecen los lenguajes formales definidos por Chomsky y se especifica un algoritmo para determinar las expresiones en los lenguajes formales asi como los autómatas finitos para reconocer cuando tales expresiones pertenecen a un lenguaje en particular. Posteriormente se formula la teoría de conjuntos borrosos definida por Zadeh.

La segunda parte se refiere a los fundamentos y filosofía de los lenguajes borrosos haciendo hincapié en la semántica y la sintáxis y su interrelación con los lenguajes naturales y formales.

Finalmente se plantea el desarrollo de un método de aprendizaje y enseñanza de un lenguaje borroso elaborado en el artículo "learning of fuzzy formal languaje" por Shinichi Tamura y Kokichi Tanaka.

2.1 LENGUAJE NATURAL

Los lenguajes naturales son un conjunto de sonidos y signos que mediante una cierta sistematización permite la comunicación entre las personas, esta sistematización toma las características de un código con lo cual aproxima de manera directa al sonido o al símbolo con la idea, es decir relaciona esencialmente al símbolo y al concepto.

El estudio de los lenguajes naturales se hace generalmente a partir del desglozamiento de sus componentes en unidades por ejemplo, sonido, palabras, locución, oración, etc., y de la correlación de estas unidades con uno o más significados, estas unidades tienen por objeto constituir la <<forma>> y el <<significado>> o la <<expresión>> y el <<contenido>> de un lenguaje. A su vez los sonidos de una lengua son estudiados por la fonología, la forma de las palabras y de como estas se combinan por la gramática y el contenido de las palabras o las unidades compuestas por estas por la semántica.

En el desarrollo de este trabajo se plantearán algunos de los resultados obtenidos en el tratamiento de la sintáxis y la semántica mediante los lenguajes formales y los lenguajes borrosos por lo tanto en este inciso sobre lenguajes naturales unicamente se mostrarán las características principales de la sintáxis y la semántica así como los problemas más importantes que de estos aspectos se derivan, omitiéndose el estudio de la fonología.

SINTAXIS

Las unidades fundamentales de la sintáxis son <<la palabra>> y la <<oración>> y se considera la sintáxis como el conjunto de reglas para combinar palabras con el objeto de formar oraciones. Donde <<la palabra>> es definida como la <<mínima forma libre>>, es decir como la forma que nunca aparece sola como expresión completa y la <<oración>> es definida como la forma lingüística independiente no incluida en una forma linguística mayor, es decir como la unidad más grande de descripción gramatical, por ejemplo las expresiones, ¡hola hace un día magnífico! ¿va de pesca?, son dos formas que pueden tener cierta conexión, sin embargo no hay una organización gramatical que las una en una más grande.

A continuación haciendo uso de la definición sobre sintáxis más arriba expuesta, se plantean los principales problemas a que esta se enfrenta para la descripción del lenguaje, y que fundamentalmente son los siguientes: La <<aceptabilidad>> y el número practicamente ilimitado de oraciones de un lenguaje.

La noción de <<aceptabilidad>> proviene de las expresiones que son o podrían ser producidas por una persona en un determinado contexto y que son o podrían ser aceptadas por otras personas de la misma lengua, aquí el problema de la lingüística, reside principalmente en especificar cuales son las oraciones aceptables, con arreglo a alguna gramática sobre la estructura del lenguaje. La importancia de la <<aceptabilidad>> en sintáxis, no solo es el de destacar la relación que existe entre un conjunto de palabras en la lengua y su mecanismo de control para explicar las ideas, hechos, etc. sino también poner de relieve las expresiones que han tenido lugar en el pasado o que se presenten en la actualidad o en el futuro.

El segundo problema reside en que toda persona es capaz de producir no solo las oraciones que alguna vez ha oido antes, sino también un número ilimitadamente grande de oraciones nuevas que jamás se le hayan presentado, en otras palabras la dificultad estriba en poder identificar las oraciones de un lenguaje cuyo número en cualquier lengua es ilimitado. Por lo tanto la gramática debe establecer las reglas capaces de incluir el conjunto indefinidamente grande de expresiones que constituyen una lengua.

Los dos aspectos anteriores, la <<aceptabilidad>> y el número casi <<ilimitado>> de oraciones en una lengua son expuestos más ampliamente a la luz de las gramáticas formales que son el tema del inciso siguiente.

A continuación se plantea en forma general la semántica de los lenguajes naturales.

SEMANTICA

La semántica es definida como el estudio del significado, sin embargo el concepto "significado" plantea una serie de problemas por la gran variedad de aspectos que cubre en el lenguaje. La intención de escribir en este inciso sobre la semántica es plantear algunas de sus principales características, así como destacar los problemas más importantes derivados del significado y que serán analizados posteriormente bajo el estudio de los lenguajes borrosos, en el capítulo correspondiente.

La referencia es una de las representaciones principales del significado y consiste de la relación que se establece entre las palabras y las cosas, hechos, acciones, o cualidades que representan. Esta relación es indirecta en el sentido en que la forma (la palabra) se relaciona con su referente a travez de su significado (concepto), sin embargo aquí existen dos dificultades debido a que cada forma no tiene asociado un solo significado ni cada significado tiene asociado una sola palabra, la primera dificultad es clasificada como: sinonimia y la segunda como : homonimia o significado múltiple, cada una de estas dificultades tiene la característica de ser indeterminadas y arbitrarias por ejemplo: las discrepancias en la clasificación de diccionarios.

Otra categoria muy importante en semántica "de conexión de significado" es la antonimia, "antagonismo en significado" por ejemplo, amor : odio, caliente : frío, y una de las relaciones más importantes es la complementaridad, ejemplo casado : soltero, macho : hembra, con la característica de que la negación de una de las partes implica la aserción de la otra, de esta manera, juan no es casado implica, juan es soltero, sin embargo pueden ocurrir situaciones en que no se implique necesariamente la negación de su complemento, por ejemplo utilizando las partículas "más o menos", se puede decir que una persona está más casada que ... otra determinada (implicando que su conducta se ajusta mejor a lo que "normalmente" caracteriza a las personas casadas). Aun cuando esto no se da con frecuencia debe ser considerado como una posibilidad dentro de la semántica. Otro aspecto de la antonimia es lo "contrario" por excelencia, grande : pequeño, joven : viejo ,y que son regularmente graduables en el sentido, de que dos cosas, pueden compararse con relación con una propiedad determinada de modo que esta "propiedad" se predique de una en un grado mayor que la otra, por ejemplo nuestra casa es mayor que la tuya, dos estados de la misma cosa puden compararse respecto a la "propiedad" en cuestión, por ejemplo, nuestra casa es mayor que hace 10 años era; una característica importante de este tipo de relaciones es que si dos términos X y Y son antonimos, entonces las oraciones que contengan X tanto implicarán como serán implicadas por las las oraciones que contengan Y, por ejemplo, nuestra casa solía ser mayor que la tuya, implica como es implicada por tu casa solía ser menor que la nuestra. Finalmente se considera otra característica importante de los antonimos a saber: Los antonimos "implícitamente" graduados y consisten de aquellas oraciones en que la negación de una no implica la aserción de la otra por ejemplo, nuestra casa no es grande no implica nuestra casa es pequeña, la importancia de esta característica es que son meras etiquetas léxicas para graduar "más que" o "menos que" con relación a una norma implícita.

El contexto es otra de las situaciones en las cuales se manifiesta el significado. El contexto en que se coloca una palabra es determinante muchas veces para responder por su significado y es limitado por indicaciones por ejemplo podrido (malo en caso de huevos,etc.), rancio (malo en caso de leche, etc.) En esta clase también existe dificultades para determinar el significado debido principalmente a la diversidad de formas (palabras).

Otro punto de vista para obtener el significado de las palabras es mediante su uso. Aun cuando el término "uso" es a su vez no muy claro, es empleado para establecer relaciones de diverso tipo en cuanto al "uso" de las palabras, un principio importante que ilustraría más ampliamente lo anterior es el siguiente: " no hay que buscar el significado de una palabra hay que buscar su uso".

Las partes anteriormente señaladas sobre semántica unicamente pretenden ilustrar algunos aspectos sobre la complejidad de establecer el significado de las palabras o las expresiones y tienen por objeto presentar de una manera global la importancia de poder determinar con más precisión el significado vía lenguajes borrosos.

1.2 LENGUAJES FORMALES

En 1960 John Backus desarrolla un método formal para definir los lenguajes de programación mediante la creación de una notación formal, que utiliza para la descripción de la sintaxis del algol 60.

Antes y en forma independiente Noam Chomsky en el año de 1956 desarrolla un modelo formal basado en un sistema de producciones gramaticales que consisten de varios tipos de gramáticas con una cierta jerarquización, uno de estos tipos de gramáticas son las denominadas libres del contexto, que contienen precisamente, a la definición formalizada por Backus para la descripción del algol 60.

Los modelos desarrollados por John Backus y Noam Chomsky proporcionan la estructura que permite la producción simple y sin ambigüedad de las reglas para la elaboración de programas, mediante un lenguaje de programación. y ha permitido enfrentarse a problemas que han surgido en teoría de autómatas y lingüística.

Para nuestro estudio de los lenguajes formales utilizaremos la notación de Noam Chomsky. En el apéndice "A" compararemos las formas notacionales con respecto a las de John Backus.

Un método práctico para analizar las oraciones en español muy utilizado son los diagramas jerárquicos denominados árboles de análisis gramatical, donde a través de las partes de la oración: frase nominal(FN) y frase verbal(FV) se obtiene la expresión lingüística resultante. Por ejemplo en el árbol de análisis gramatical, representado en la fig. 2.1, vemos las partes de la gramática asociada a los elementos de la oración. En este árbol de análisis gramatical hay dos tipos de palabras las que se encuentran dentro del rectángulo llamadas terminales y el otro tipo denominadas no-terminales. Aparentemente puede pensarse que este árbol de análisis fue derivado de una definición formal del español, es en realidad derivado de un conjunto heurístico de reglas que vienen aplicándose a lo largo de muchos años. Sin embargo se presentará el método como se maneja un árbol de análisis gramatical formal y como es construido un árbol de análisis gramatical relativo a una gramática formal.

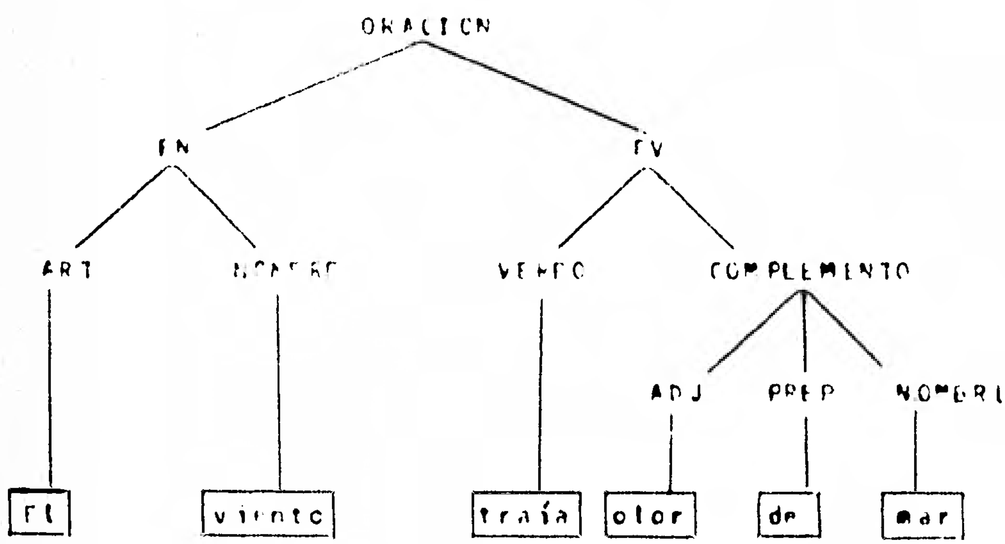


Fig. 2.1 Arbol de análisis gramatical.

Se continuará planteando tres problemas que son: la especificación, determinación y reconocimiento de los lenguajes formales y que serán desarrollados por separado.

2.2.1 ESPECIFICACION DE LOS LENGUAJES FORMALES

El problema de la especificación de los lenguajes surge principalmente en el desarrollo formal de los lenguajes y es resuelto mediante la jerarquización de Chomsky.

Se iniciará con algunas definiciones básicas que son necesarias para la especificación de la estructura de las gramáticas y los lenguajes formales.

Un "alfabeto" es un conjunto finito de símbolos, correspondientes a los símbolos terminales de un árbol de análisis gramatical, en el español este estaría constituido por las 29 letras, los símbolos de puntuación y un blanco. En un lenguaje formal serían símbolos individuales.

Una "cadena" es una sucesión de símbolos concatenados linealmente, elegidos del alfabeto del lenguaje.

Un lenguaje formal está constituido por ciertas reglas también llamadas gramáticas o reglas sintagmáticas, que especifican las cadenas aceptadas por el lenguaje.

Ejemplo 2.1 Generación de un lenguaje formal.

Para generar un lenguaje formal se da un alfabeto sea en este caso el conjunto de símbolos (a,b,c,*) . Un lenguaje usando este alfabeto puede tener informalmente las siguientes reglas gramaticales:

1) Una expresión es una cadena de dos o más símbolos.

2) Una expresión es aceptada por el lenguaje si la cadena está precedida por el símbolo "*".

Una cadena aceptada por tal lenguaje sería "*a*ab" en donde las comillas solo indican la separación de una expresión a otra, mediante un árbol de análisis gramatical (fig. 2.2) se representan la cadena aceptada por tal lenguaje.

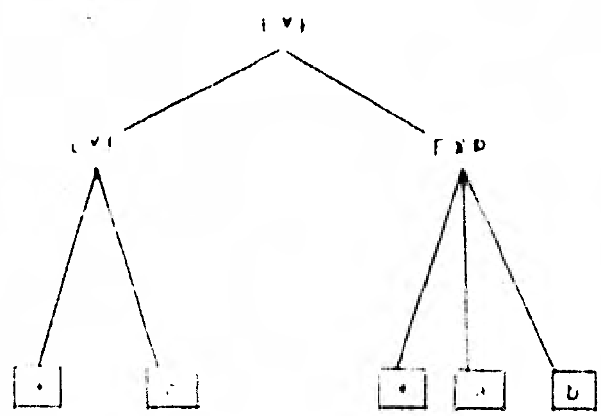


Fig. 2.2 Arbol de análisis gramatical de un lenguaje formal.

Las cadenas de símbolos de un alfabeto están sujetas para su manipulación por un álgebra, mediante la cual se pueden llevar a cabo algunas operaciones de gran utilidad.

La "longitud" de una cadena w se define como el número de símbolos concatenados linealmente y se representa por long(w). La "cadena vacía" es la cadena de longitud cero y se designa por e, es decir long(e) = 0.

Un conjunto de la forma $\{a^n b^n \mid n \geq 0\}$ significa el conjunto de todas las cadenas de longitud n (finita) de "ab", donde n va tomando valores positivos desde cero de la siguiente manera:

n = 0	e	la cadena nula
n = 1	ab	
n = 2	aabb	
n = 3	aaabbb	
n = 4	aaaabbbb	
	....	
n = m	aa.... aabb.... b	
	m-veces m-veces	

Conceptos de Lenguajes Formales y Aplicaciones

Una operación que se aplica con mucha utilidad a las cadenas de un conjunto de caracteres en un alfabeto, llamada cerradura Kleene y representada por * consiste de la cadena vacía y todas las cadenas compuestas por los caracteres de el alfabeto.

Ejemplo 1.- Cerradura Kleene.

El lenguaje {0,1}* consiste de {e,0,1,00,01,10,11,000,...}

Otra operación representada por + y muy útil en el manejo de cadenas consiste de las cadenas compuestas por los caracteres del alfabeto excepto la cadena vacía, ejemplo de esta operación ser fue, { 0,1 }+ genera el lenguaje formado por el conjunto de cadenas siguientes {0,1,00,01,10,11,000,...}

REGLAS SINTAGMATICAS O GRAMATICAS DE LOS LENGUAJES FORMALES

Para generar los lenguajes se deben especificar las reglas mediante las cuales se produzcan las cadenas válidas del lenguaje.

Noam Chomsky proporcionó una definición y una clasificación de las reglas sintagmáticas de los lenguajes formales a las que denominó gramáticas tipo 0, tipo 1, tipo 2, tipo 3 donde cada tipo de gramática genera un conjunto de lenguajes de tal manera que las gramáticas o lenguaje tipo 0 comprenden en su definición a las tipo 1 y estas a su vez comprenden a las tipo 2 y estas a las tipo 3.

En teoría de lenguajes formales una regla sintagmática o gramática esta constituida por cuatro entidades a saber:

1) Un alfabeto finito de símbolos terminales designado por Vt.

2) Alfabeto finito de símbolos no-terminales designado por Vn.

3) Un conjunto de producciones o reglas de reescritura designadas por P y se representan mediante una flecha (--->).

4) Un símbolo inicial no-terminal designado por S.

Con las siguientes características $V_t \cap V_n = \emptyset$ la intersección del conjunto de símbolos terminales y el conjunto de símbolos no-terminales es vacía y la unión de tales conjuntos (se denota por $V = V_n \cup V_t$) constituye el conjunto de símbolos de la gramática. El conjunto de producciones son las reglas de reescritura que permiten que los símbolos terminales y no terminales sean combinados para formar el lenguaje. El símbolo inicial $S \in V_n$ también llamado raíz del lenguaje proporciona el punto de partida de las producciones para generar las cadenas del lenguaje estos elementos conforman un cuádruple denotado por $G = (V_t, V_n, P, S)$.

Habiendo definido las gramáticas, el siguiente paso es establecer los lenguajes generados por tales gramáticas, para tal efecto se considera la siguiente simbología, dada una gramática $G = (V_n, V_t, P, S)$ las relaciones $\underset{G}{\Longrightarrow}$ y $\overset{*}{\underset{G}{\Longrightarrow}}$ entre las cadenas en V^*, significan lo siguiente, si $W_1 \longrightarrow W_2$ es una producción en P y V_1, V_2 son cadenas en V^* entonces $V_1W_1V_2 \longrightarrow V_1W_2V_2$ significa que la producción $W_1 \longrightarrow W_2$ es aplicada a la cadena $V_1W_1V_2$ para obtener $V_1W_2V_2$, de esta manera $\underset{G}{\Longrightarrow}$ relaciona dos cadenas dentro de la gramática.

donde la segunda es obtenida de la primera por la aplicación de una sola producción. Ahora si $w_1, w_2, w_3, \ldots, w_m$ son m cadenas

en V^* y $w_1 \underset{G}{\Longrightarrow} w_2, w_2 \underset{G}{\Longrightarrow} w_3, \ldots, w_{m-1} \underset{G}{\Longrightarrow} w_m$ entonces

$w_1 \overset{*}{\underset{G}{\Longrightarrow}} w_m$ significa que podemos obtener w_m de w_1 mediante la

aplicación de un número finito de producciones de P.

El lenguaje generado por G es L(G) y consiste del conjunto de todas las cadenas que son obtenidas mediante la aplicación de las producciones que caracterizan a la gramática esto formalmente se puede escribir como:

$L(G) = \{ w \in V_t^* \mid$ existe una derivación $S \underset{G}{\Longrightarrow} \ldots \underset{G}{\Longrightarrow} w$ esta en G $\}$

donde $V_t^* = (V_t \cup \{ e \})$. Es decir si de las cadenas generadas por las producciones P de G se puede derivar al menos una cadena W que consista unicamente de símbolos terminales entonces el conjunto de las W es el lenguaje L(G) generado por G.

Las reglas de producción generan un conjunto de cadenas válidas en el lenguaje a partir del símbolo inicial. En cada cadena generada se van reemplazando los símbolos de Vn encontrados, continuando el proceso hasta que en la cadena solo se encuentren símbolos terminales (símbolos de Vt) al llegar a esta etapa del desarrollo la generación termina y la cadena pertenece al lenguaje.

Ejemplo 2.7 Definición de un lenguaje formal.

El lenguaje definido en la ejemplo 2.1 Puede formalmente ser definido de la siguiente manera:

```
 dada la gramática G = (Vt,Vn,P,S)
donde Vn = { S,A,B,E }
      Vt = { +,a,b,c }
       P = 1         ( S
           2           S ---> SE
           3           S ---> +
           4           E ---> SE
           5           E ---> A
           6           A ---> a
           7           A ---> AB
           8           B ---> BC
           9           B ---> b
           10          BC ---> C
           11          C ---> c )
```

esta gramática esta constituida por cuatro símbolos terminales (+,a,b,c), cuatro símbolos no-terminales (S,A,B,E), el símbolo inicial S y once reglas gramaticales mediante las cuales se establecen las cadenas del lenguaje que pueden ser construidas, para obtener la cadena "+a+ab", se comienza aplicando el símbolo inicial y reemplazando los elementos no-terminales en la cadena como lo indiquen las reglas de la gramática:

Regla	cadena
1	S
2	SE
2	SEE
4	SESE
1	•ESE
5	•ASE
3	•A•E
5	•A•A
6	•A•AB
7	•a• AB
8	•a•aB
9	•a•ab

Para representar el proceso anterior utilizamos el árbol de análisis gramatical representado en la fig. 2.3

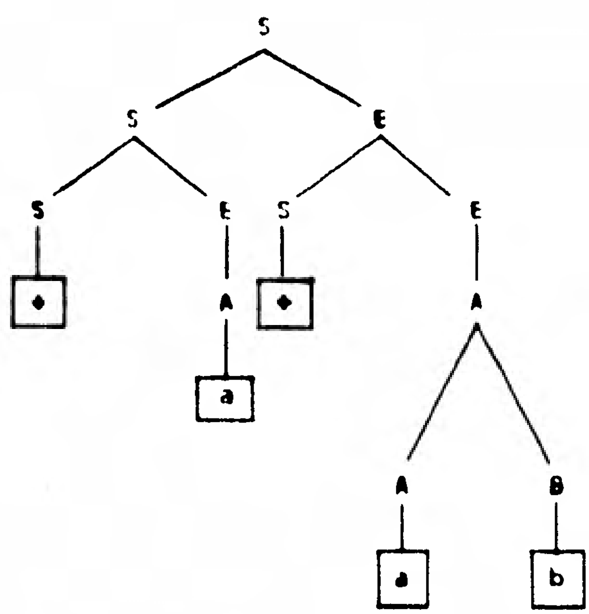


Fig. 2.3 Derivación de la cadena •a•ab.

El análisis se realiza tomando en consideración las siguientes reglas:

Si el objeto S representa un elemento de V = (Vn U Vt) entonces un árbol con raíz S es una de las dos cadenas.

1) Si S es miembro de Vt, entonces S encerrado en un rectángulo es una hoja del árbol o terminal.

2) Si S es miembro de Vn entonces S tiene una sucesión de ramas o subárboles como hijos que son obtenidos mediante las reglas de producción.

2.2.1.1 GRAMATICAS TIPO 3

Las gramáticas definidas por Noam Chomsky, esto es las gramáticas tipo 3, tipo 2, tipo 1 y tipo 0 son diferenciadas mediante un conjunto de restricciones que vienen a caracterizar las producciones que las conforman, de esta manera, las gramáticas tipo 3 llamadas también gramáticas regulares tienen reglas de dos formas:

1) Un símbolo no-terminal es reescrito como un símbolo terminal.

2) Un símbolo no-terminal es reescrito como un símbolo no-terminal seguido de un símbolo terminal.

esto es, si $G = (V_n, V_t, P, S)$ es una gramática y suponemos todas las producciones de la forma:

$A \dashrightarrow a$

o

$A \dashrightarrow a\,B$

donde A, B son miembros de V_n y a pertenece a V_t entonces G es una gramática regular o tipo 3.

El lenguaje generado por las gramáticas tipo 3 se denominan lenguajes tipo 3 o regulares.

Ejemplo 2.4 Lenguaje generado por una gramática tipo 3.

Un lenguaje de este tipo es { a^n | $n \geq 1$ } donde la gramática tipo 3 que lo genera puede ser establecida de la siguiente forma G = (Vn,Vt,P,S) donde

Vn = { S }
Vt = { a }
S = { S }
P = { S ---> Sa
S ---> a }

así para n = 4 tenemos la cadena aaaa que puede ser representada mediante el árbol de análisis gramatical de la fig. 2.4

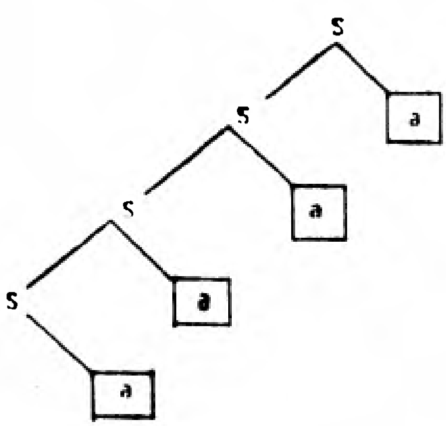


Fig. 2.4 Lenguaje generado por una gramática tipo 3.

2.2.1.2 GRAMATICA TIPO 2

Las gramáticas tipo 2 también llamadas libres del contexto generan los lenguajes libres del contexto o tipo 2. Estas gramáticas tipo 2 estan caracterizadas por producciones de la forma:

A ---> w

donde A es un solo elemento de Vn y W es cualquier cadena de (Vn U Vt)* de esta manera A puede ser reemplazada por la cadena W independientemente del contexto en que aparezca.

Ejemplo 2.5 Lenguaje generado por una gramática tipo 2.

Un lenguaje generado por esta gramática sería el siguiente: { a^n b^n | n $\geq$ 1 } donde la gramática estaría constituida por G = (Vn = {S} ,Vt = {a,b} ,S ={ S },P ={ S ---> aSb, S --->ab}) así para n = 4 tenemos la cadena aaaabbbb cuyo árbol de análisis gramatical esta desarrollado en la fig. 2.5

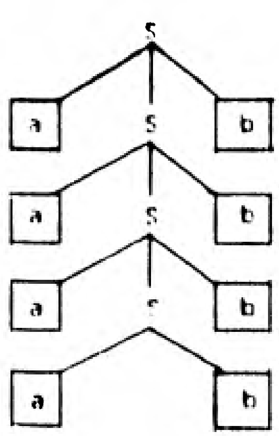


Fig. 2.5 lenguaje generado por una gramática tipo 2.

2.2.1.4 GRAMATICAS TIPO 1

Las gramáticas tipo 1 o sensibles de contexto generan los lenguajes sensibles del contexto y permiten menos restricciones en la definición de sus producciones, por lo cual comprenden a las gramáticas tipo 2 y tipo 3 y son representados por G = (Vn,Vt,P,S) donde

W1 ---> W2

son producciones en que long(W1) ≥ long(W2) es decir, la longitud de W1 en símbolos debe ser mayor que o igual que la longitud de W2 también pueden representarse las gramáticas sensibles del contexto de la siguiente forma:

W1AW2 ---> W1BW2

donde las W"s pertenecen a (V_n U V_t)*, A pertenece Vn y B pertenece a (Vn U Vt)+, es de notar que A será reemplazada por B siempre que A aparezca en el contexto de W1, W2.

Ejemplo 2.6 Lenguaje generado por una gramática tipo 1.

un lenguaje generado por la gramática tipo 1 sería { $a^n b^n c^n$ | n ≥ 1 } donde G = (Vn,Vt,P,S) y

Vn= (S, A, B, C, D)

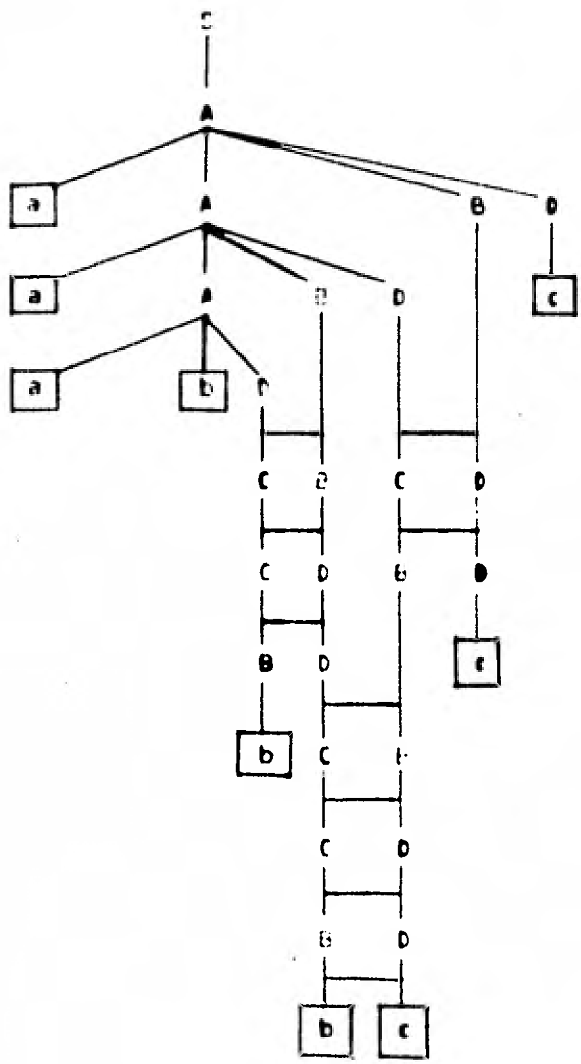
Vt = (a, b, c)

S = (S)

```
P =( S ---> A  A ---> aABD
     A ---> abD
     DB ---> CB
     CB ---> CD
     CD ---> BD
     bB ---> bb
     D ---> c )
```

así para n = 3 tenemos la cadena aaabbbccc cuyo árbol de análisis gramatical esta desarrollado en la fig. 2.6

Fig 2.6 Lenguaje generado por una gramática tipo 1.



2.2.1.4 GRAMATICAS TIPO 0

Las gramáticas tipo 0 permiten cualquiera de las reglas que caracterizan a las gramáticas tipo 3, tipo 2 y tipo 1 y pueden ser representadas mediante producciones de la forma:

w1 ---> w2

donde w1 es una cadena que pertenece a (Vn U Vt)+ y w2 pertenece a (Vn U Vt)* este tipo de lenguaje son también llamados lenguajes numerables recursivamente y generan los lenguajes tipo 0, como ejemplos de lenguajes generados por este tipo de gramáticas se pueden considerar los ejemplos 2.4,2.5,2.6.

2.2.2 DETERMINACION DE LOS LENGUAJES FORMALES

Habiendo especificado las gramáticas de los lenguajes formales el siguiente paso es hallar el proceso mediante el cual se pueda resolver el problema de determinar cualquier sentencia en un lenguaje formal, para tal efecto se recurre a un algoritmo, que será planteado después de las observaciones siguientes:

Un lenguaje cuyas sentencias puedan ser generadas por medio de un procedimiento se denominan recursivamente numerables, es decir, si existe un procedimiento mediante el cual se reconozcan las sentencias en el lenguaje.

Un lenguaje se denomina recursivo si existe un algoritmo que reconozca al lenguaje, es decir para cualquier sentencia W, ¿cuando W esta en el lenguaje?.

Ahora si G = (Vn,Vt,P,S) es una gramática sensible del contexto o tipo 1 entonces por definición, la sentencia e puede estar en L(G) (el lenguaje generado por G) si y solo si P contiene la producción S ---> e. Si tal producción es eliminada (si existe) se puede formar una nueva gramática tipo 1 o sensible del contexto G1 = (Vn,Vt,P,S) la cual genera L(G) - (e) y la longitud del lado derecho es al menos tan grande como el lado izquierdo (ya que long(e) = 0) ademas las formas sentenciales son no decrecientes. Si V = Vn U Vt tiene K simbolos, sea W e Vt* donde

$$S \overset{*}{\underset{G1}{\Longrightarrow}} W, \text{ siendo esta derivación } S \Longrightarrow A1, \ldots \Longrightarrow An = W ,$$

tenemos que la long(A1) $\leq$ long(A2) $\leq \ldots \leq$ long(An) supongamos ahora Ai,...Ai+j tienen la misma longitud, y sea esta longitud igual a l y también sea H $\geq K^l$, entonces dos de las Ai,...,Ai+j deben ser las mismas ya que unicamente hay K^l cadenas de longitud l en V, por lo cual se puede omitir un paso en la derivación de W, entonces, existe otra derivación menor para obtener esta misma W.

Ahora para poder reconocer un lenguaje formal debemos desarrollar un procedimiento que nos permita obtener las sentencias que lo constituyan y a tal procedimiento darle el carácter de algoritmo, y mediante este algoritmo determinar cuando una sentencia esta en el lenguaje.

Sea $G = (V_n, V_t, P, S)$ una gramática sensible del contexto, por las observaciones anteriores si e esta en el lenguaje generado por tal gramática puede ser eliminada. Se supone entonces que P (las producciones) no contienen a la producción $S \text{ ---> } e$ y sea una cadena W en V_t^*, tal que $long(w) = n$. Se define el conjunto T_m como el conjunto de cadenas $\$$ en V_t^*,

de longitud al menos n y tales que $S \overset{*}{\underset{G}{\Longrightarrow}} \$$ por

alguna derivación de al menos m pasos, entonces $T_m = T_{m-1} \cup \{ \$ \mid$ para alguna B en T_{m-1} donde $B \Longrightarrow \$$ y $long(\$) = n$) Es decir, si $S \overset{*}{\Longrightarrow} \$$ y $long(\$) = n$ entonces $\$$ esta en T_m para alguna m o si S no deriva a $\$$ o $long(\$) > n$, entonces $\$$ no esta en T_m para cualquier m y $T_{m-1} \subseteq T_m$ para toda $m \geq 1$, ya que T_m depende de T_{m-1}, además si $T_m = T_{m-1}$, entonces $T_m = T_{m+1} = T_{m+2} = \ldots$ por lo que el algoritmo se hara para calcular $T_1, T_2, T_3, \ldots$, hasta alguna m, en la cual se llegue a que $T_m = T_{m-1}$, en dos etapas se mostrara que tal algoritmo es factible.

Primera. Si W no esta en T_m entonces no esta en el lenguaje generado por G ya que para $j > m$ tenemos que $T_j = T_m$. Es decir, si W esta en T_m

entonces $S \overset{*}{\underset{G}{\Longrightarrow}} W$.

Segunda. El procedimiento tiene terminación es decir, para alguna m, $T_m = T_{m-1}$, se ha visto que para $i \geq 1$, $T_{i-1} \subseteq T_i$. Si $T_i \neq T_{i-1}$ entonces el número de cadenas del lenguajes en T_i es al menos una mayor que el número de T_{i-1} pero si $V = (V_n \cup V_t)$ tiene k símbolos, entonces el número de cadenas en V^* de longitud menor que o igual a n es $k^1 + k^2 + k^3 + \ldots + k^n$, serie cuyo resultado es menor que o igual que $(k + 1)$ por lo tanto estas son todas las cadenas que pueden estar en T_i, para alguna i. Con lo cual tenemos que $T_m = T_{m-1}$ para alguna $m \leq (k + 1)^{n+1}$. Es decir cuando dos conjuntos son hallados iguales mediante el procedimiento anterior se ha terminado y determinado si tal o cual cadena esta en el lenguaje o no lo esta.

Ejemplo 2.7 Proceso para determinar las sentencias en un lenguaje.

Se considera la gramática del ejemplo 2.6 cuyas producciones son:

P = 1.- { S ---> T
2.- T ---> aTBD
3.- T ---> abD
4.- DB ---> BD
5.- bB ---> bb
6.- bD ---> bc
7.- cB ---> Bc
8.- D ---> c }

para determinar si W1 = aabbcc esta en el lenguaje generado por esta gramática recurrimos al algoritmo anterior. La primera cadena en T0 es el símbolo inicial, la segunda producto de la aplicación de la regla 2 constituye T1 L T2.

T0 = { S }
T1 = T0 U {1}

la tercera y cuarta cadena vienen de T1 y la aplicación

T2 = T1 U {aBCD, abC}

se continua aplicando las reglas 3, 8, (2 y 8), 8 a las cadenas de T2 para obtener T3.

T3 = T2 U {aabDBD, aBCc, aabDBc, abc}

para obtener T4 aplicamos la reglas 4, 8, 9, 6, 8 a cada una de las cadenas de T3. Es de notarse que en T4 ya no aparece la cadena abc debido a que esta constituida unicamente de símbolos terminales por lo que ya no tendrá importancia sobre el resultado final de la cadena W1.

T4 = T3 U {aabCBD, aabcBD, aabCBc, aabcBc}

T5 es obtenido mediante la aplicación de las reglas 5, 8, (5 y 8) a las cadenas de T4.

T5 = T4 U {aabCDD, aabCBc, aabCDc}

T6 es obtenido mediante la aplicación de las reglas 6 y 8.

T6 = T5 U {aabCDD, aabCcD, aabCcc}

finalmente para obtener T_7 aplicamos las producciones (7 y 8), lo cual nos permite obtener w_1 = aabbcc.

$$T_7 = T_6 \cup \{aabbcc, aabbcD, aaBbcD, aaBbcc, aabBDc\}$$

además es claro que cualquier cadena de símbolos terminales de a"s, b"s, c"s diferente de w_1 no se encuentra en el lenguaje debido a que $T_7 = T_8 = T_9 = \ldots$

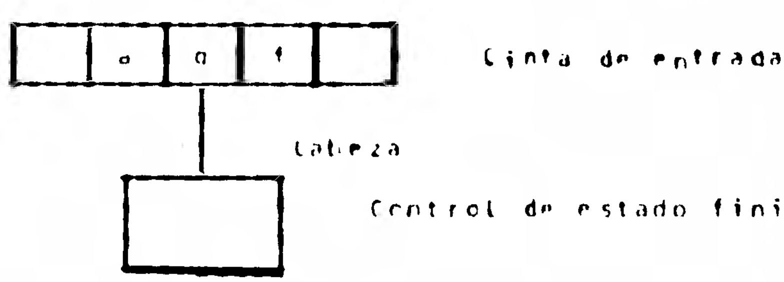
2.2.7 RECONOCIMIENTO DE LOS LENGUAJES FORMALES.

Habiendo planteado las soluciones a los problemas de la especificación y la determinación de los lenguajes formales, el siguiente paso es establecer el mecanismo que determine a partir de la definición de un lenguaje, el tipo de lenguaje a que pertenece dentro de la jerarquía de Chomsky, es decir si el lenguaje definido es tipo 3, tipo 2, tipo 1 o tipo 0, para lo cual recurrimos a la teoría de autómatas o reconocedores, que nos proporcionan el medio de caracterizar los lenguajes dentro de la jerarquía antes señalada.

AUTOMATAS DE ESTADO FINITO

Un automatón finito determinístico es una máquina que reconoce cadenas de símbolos y esta constituida por una cinta de entrada dividida en cuadros, una cabeza de lectura a la cinta y un control de estado finito fig. 2.7

fig. 2.7 Autómata finito



Un automatón finito determinístico es denotado por un quíntuple $M = (K, \Sigma, \delta, q_0, F)$ donde:

1.- K es el conjunto de estados del control finito.

2.- Σ es el alfabeto de entrada en la cinta y cada cuadro contiene un símbolo.

3.- δ Es una función de transición de $K \times \Sigma$ en K, tal que si $\delta(Q,X) = P$, entonces cada vez que M se encuentra en el estado Q con el símbolo X sobre la cabeza de entrada, M mueve la cabeza de entrada a la derecha y pasa al estado P.

4.- Q_0 es un estado distinguido en K y representa al estado inicial.

5.- $F \subseteq K$ es un conjunto de estados finales o estados aceptados.

El dominio de la función de transición δ de $K \times \Sigma$ en K puede ser extendido a $K \times \Sigma^*$ definiendo $\hat{\delta}$ como sigue:

$$\hat{\delta}(Q,\varepsilon) = Q$$

$$\hat{\delta}(Q,XA) = \delta(\hat{\delta}(Q,X),A) \text{ para cada } X \text{ en } \Sigma^* \text{ y } A \text{ en } \Sigma$$

$\delta(Q,X) = P$ significa lo siguiente, cuando el automatón finito M se encuentra en el estado Q y la cadena X en la cinta de entrada, entonces M pasará al estado P, cuando la cabeza de entrada del control finito se mueva a la derecha de la cadena X.

Se dice, que una sentencia X es aceptada por un automatón finito determinístico M, si $\delta(Q_0,X) = P$, para cualesquiera P en F (es decir el automatón finito determinístico aceptara la cadena X si a partir de un estado inicial y analizando la cadena X el automatón M pasa a un estado final P). El conjunto de todas las cadenas X aceptadas por el automatón finito determinístico M se le designa por T(M) esto es $T(M) = \{ X \mid \delta(Q_0,X)$ esta en $F \}$. Un autómata finito determinístico que acepte cualquier conjunto de cadenas es denominado autómata finito regular. Los autómatas para su manipulación son representados mediante un diagrama de estados que consiste de una serie de círculos que indican los estados y tales que dos estados Q y P cualesquiera en K estan unidos mediante una flecha etiquetada con un símbolo X de Σ, si el automatón finito determinístico en el estado Q y analizando la cadena X puede pasar al estado P, además los estados finales estan caracterizados por un doble circulo y el estado inicial mediante una flecha etiquetada con la palabra "inicial".

Ejemplo 2.2 Autómata finito determinístico.

Especificamos primero el automatón finito determinístico M de la siguiente forma:

M = (K,Σ,δ,q_0,F) donde

$\Sigma = \{ 0,1 \}$

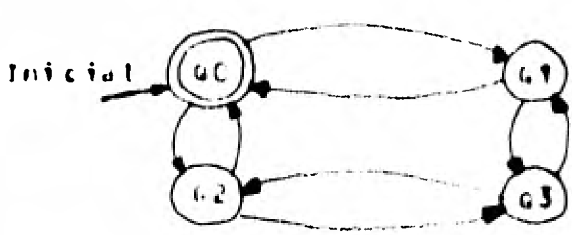
$K = \{ q_0,q_1,q_2,q_3 \}$

$F = \{ q_0 \}$

Y cuyas funciones de transición son:

1.- $\delta(q_0,0) = q_2$
2.- $\delta(q_1,0) = q_3$
3.- $\delta(q_2,0) = q_0$
4.- $\delta(q_3,0) = q_1$
5.- $\delta(q_0,1) = q_1$
6.- $\delta(q_1,1) = q_0$
7.- $\delta(q_2,1) = q_3$
8.- $\delta(q_3,1) = q_2$

tal autómata finito puede ser representado mediante el siguiente diagrama de estados.



Supongamos ahora que tenemos la siguiente cadena $X = 0101$ y se desea saber si tal cadena puede ser reconocida por la función de el automatón finito, aplicando la función de transición a la cadena y tenemos que.

$$\delta(q_0,0) = q_2$$
$$\delta(q_2,1) = q_3$$
$$\delta(q_3,0) = q_1$$
$$\delta(q_1,1) = q_0$$

es decir puesto que $\delta(q_0,0101) = q_0$ entonces X esta en $T(M)$ y la cadena ha sido aceptado por el automatón antes especificado.

En resumen un automatón finito determinístico consiste de un "control" y una cinta de entrada que es analizada de izquierda a derecha mediante una cabeza a la cinta, y el automaton realiza un "movimiento" determinado por el estado actual del control de estados y el simbolo sobre la cabeza a la cinta y cada movimiento se lleva a cabo a partir de un nuevo estado y del corrimiento de la cabeza de entrada un cuadro a la derecha.

Una generalización de un automatón finito determinístico es definido como un automatón finito no-determinístico tal que para cada estado y el símbolo de entrada, el automatón finito no-determinístico tiene ninguna o varias elecciones para realizar el movimiento siguiente y puede también elegir el cambio de estado sin necesidad de un corrimiento de la cabeza a la cinta.

Un automatón finito no-determinístico M es un sistema $M = (K, \Sigma, \delta, Q0, F)$ donde:

1.- K es un conjunto de estados del control.

2.- Σ es un alfabeto desde donde son elegidos los símbolos.

3.- δ es la función de transición de estados que mapea $K \times (\Sigma \cup \{ e \})$ a un conjunto de K.

4.- Q0 en K es el estado inicial del control finito.

5.- $F \subset K$ es el conjunto de estados finales o estados aceptados.

La diferencia entre un automatón finito determinístico y uno no-determinístico estriba en que la función de transición $\delta(Q,A)$ es un conjunto de estados (o el estado vacío) en vez de un solo estado.

La interpretación de $\delta(Q,A)$ en un automatón finito no-determinístico M es que $\delta(Q,A) = P1,\ldots,Pn$, es decir, M en el estado Q, analizando A en la cinta de entrada mueve la cabeza de la cinta un cuadro y elige cualquiera de los estados siguientes $P1,\ldots,Pn$.

El dominio de la función de transición puede ser extendido a $K \times \Sigma^*$ definiendo:

$$\delta(q, \varepsilon) = \{q\} \qquad \text{y}$$

$$\delta(q, xA) = \bigcup_{P \text{ en } \delta(Q,x)} \delta(P, A) \quad \text{para cada } x \text{ en } \Sigma^* \text{ y } A \text{ en } \Sigma.$$

Una sentencia x es aceptada por M si hay un estado $\delta(Q0, x) = P$ en F.

El conjunto de todas las sentencias aceptadas por M se denota por $T(M)$.

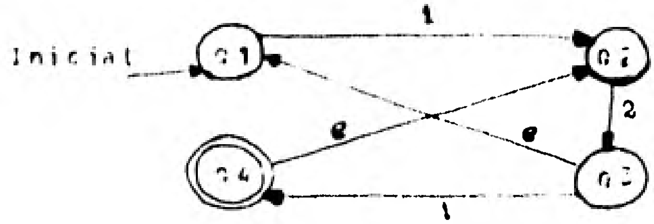
Ejemplo 2.5 Autómata finito no-determinístico.

$$M = (\{ Q1, Q2, Q3, Q4 \}, \{ 1, 2, 3 \}, \delta, Q1, \{ Q4 \})$$

Donde la función de transición δ es definida como sigue.

estado	1	2	3
Q1	{Q1, Q2}	{Q1}	∅
Q2	∅	{Q3}	∅
Q3	{Q4}	∅	{Q1}
Q4	∅	∅	{Q2}

Y cuyo diagrama de estado es el siguiente:


Inicial
1
2
e
e
1

Consideremos ahora que $y = 12121$ es la cadena de entrada y se desea saber si tal cadena puede ser reconocida por M, entonces aplicando la función de transición a Y tenemos que

$$\delta(q_1,1) = q_1$$
$$\delta(q_1,2) = q_1$$
$$\delta(q_1,1) = q_2$$
$$\delta(q_2,2) = q_3$$
$$\delta(q_3,1) = q_4$$

es decir puesto que $\delta(q_1,12121) = q_4$ que es un estado final la cadena Y es aceptada por M.

El teorema siguiente cuya demostración aparece en el apéndice A, nos permite relacionar los lenguajes generados por las gramáticas tipo 3 y los autómatas finitos.

Teorema 2.1 Si $G = (V_n,V_t,P,S)$ es una gramática tipo 3 entonces existe un autómata finito $M = (K,V_n,\delta,S,F)$ tal que $T(M) = L(G)$.

1.- M es considerado un automatón finito no determinístico.

2.- Los estados de M son los símbolos no-terminales de G.

3.- M tiene un estado adicional A ∉ Vn, es decir K = Vn U { A } .

4.- El estado inicial de M es S.

5.- Si P tiene la producción S ---> ε entonces los estados finales de M son F = { S,A } , si no ocurre lo anterior entonces F = { A }.

6.- El estado A esta en δ(B,a) si B ---> a esta en P. Es decir δ(B,a) contiene todos los estados C si B ---> aC y δ(A,a) = 0 para toda a en Vt.

El autómata de estado finito cuando acepta una sentencia X, simula una derivación de X por la gramática G.

Ejemplo 2.10 Autómata de estado finito.

Se considera la gramática G = ({S,B},{ a,b },P,S) y P esta constituido por las producciones siguientes :

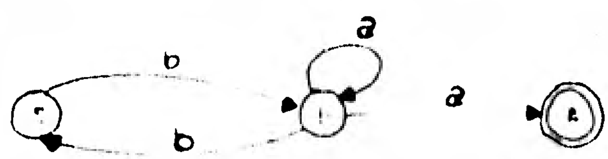
1.- S ---> aB
2.- B ---> aB
3.- B ---> bS
4.- B ---> a

se construye un automatón finito no-determinístico $M = (\{S,B,A\},$ $\{a,b\},\delta,S,$ [illegible] $)$ donde la función de transición δ esta dada por:

1.- $\delta(S,b) = \{$ [illegible] $\}$
2.- $\delta(S,b) = \emptyset$
3.- $\delta(B,a) = \{$ [illegible]$,A\}$
4.- $\delta(B,b) = \{$ [illegible] $\}$
5.- $\delta(A,a) = \emptyset$
6.- $\delta(A,b) = \emptyset$

el diagrama de estado finito del automatón sería:

Inicial

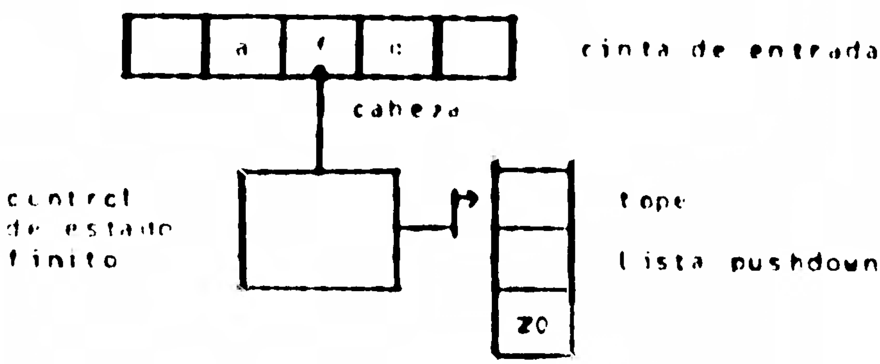


el cual nos permite reconocer todas las cadenas X constituidas por igual número de ceros precedidas a igual número de unos que es justamente $L(G)$.

AUTOMATAS PUSHDOWN

Los automatones pushdown son dispositivos que permiten reconocer los lenguajes libres del contexto y pueden considerarse como una extensión de los automatones finitos, debido a que tienen el control de la cinta de entrada y un almacenamiento pushdown, fig. 2.9

Fig. 2.9 Autómata pushdown.



El almacenamiento pushdown es una lista en la cual el primer elemento en entrar es el último en salir, esto es, los símbolos únicamente pueden ser removidos en el tope de la lista (donde el tope de la lista será el símbolo más a la izquierda).

Formalmente un automatón pushdown (M), puede ser definido como un sistema $M = (K, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ donde:

1.- K es un conjunto finito de estados de el control finito.

2.- Z es un alfabeto finito, llamado el alfabeto de entrada.

3.- T es el alfabeto de la lista pushdown.

4.- δ Es una función de transición de $K \times (Z \cup \{e\}) \times T$ al subconjunto finito $K \times T^*$.

5.- Z_0 en T es un símbolo en la lista de almacenamiento pushdown, llamado símbolo inicial.

6.- $F \subseteq K$ es el conjunto de estados finales.

7.- Q_0 en K es el estado inicial del control finito.

Una configuración de un autómata pushdown es un par (Q, W) donde Q es un estado en K y W es una cadena de símbolos pushdown. Se dice que un autómata pushdown M es una configuración (Q ,W) si M en estado Q con W sobre el almacenamiento pushdown, el símbolo más a la izquierda de W está en el tope del almacenamiento pushdown. Si a está en $Z \cup \{e\}$, w, y B están en T^* y Z_1 está en T y además si el par (P,B) está en $\delta(Q,a,Z_1)$, entonces se escribe

$$a:(Q,\ Z_1W) \vdash_M (P,\ BW)$$

lo que significa de acuerdo a las reglas del autómata pushdown, la entrada a puede causar que M vaya de la configuración (Q, Z_1W) a la configuración (P,BW).

Si para $a1, a2, \ldots, an$ cada una en $\Sigma \cup \{ e \}$, los estados $Q1, Q2, \ldots, Qn+1$ y las cadenas pushdown $W1, W2, \ldots, Wn+1$ tenemos:

$$ai : (Qi, Wi) \vdash_{M} (Qi+1, Wi+1)$$

para toda i entre 1 y n, entonces se escribe

$$a1a2 \ldots an : (Q1, W1) \overset{*}{\vdash}_{M} (Qn+1, Wn+1)$$

además para a = e se tiene por convención

$$e : (Q, W) \overset{*}{\vdash}_{M} (Q, W)$$

El autómata pushdown sera no-determinístico y tendrá un número finito de opciones para seleccionar un movimiento dependiendo de la situación. Los movimientos del automatón pushdown serán de dos tipos.

Primero. Un símbolo es analizado en la cinta de entrada donde dependiendo de este símbolo, el símbolo en el tope de la lista y el estado del control finito, un numero de elecciones son posibles y cada elección consiste de un estado siguiente para el control finito y una cadena de símbolos (posiblemente vacía) para reemplazar al símbolo en el tope del almacenamiento pushdown e inmediatamente después si se requiere la cabeza de entrada avanza un símbolo.

Segundo. Unicamente permite manipular el almacenamiento pushdown sin necesidad de que la cabeza de entrada avance ni se utilice el simbolo de entrada.

El lenguaje aceptado por un autómata pushdown puede ser definido de dos formas.

Primero. El lenguaje aceptado será el conjunto de todas las entradas para las cuales se lleve a cabo una sucesión de movimientos y el automatón pushdown M obtenga su almacenamiento pushdown vacío. Se dice que un lenguaje, es un lenguaje aceptado por almacenamiento vacío, y se denota por N(M) y será

$$\{ W \mid W : (q_0, Z_0) \vdash^{*}_{M} (q, e) \text{ para cualquier } q \text{ en } K \}$$

cuando acepta por almacenamiento vacío, el conjunto de estados finales es irrelevante. Así cuando acepta por almacenamiento vacío usualmente el conjunto de estados finales sera el conjunto vacío.

Segundo. El lenguaje aceptado será el conjunto de todas las entradas para las cuales se lleve a cabo una sucesión de movimientos y el automatón pushdown M entre a un estado final. El lenguaje se dice que es aceptado por estado final y se define por T(M) y será

$\{W \mid W : (Q0, Z0) \overset{*}{\underset{M}{\vdash}} (Q,W)$ para cualesquiera W en T* y Q en F $\}$

Ambas definiciones son equivalentes ya que si un conjunto puede ser aceptado por almacenamiento vacío por un automatón pushdown también será aceptado por estado final.

La función de transición.

$$\delta : K \times (Z \cup \{ e \}) \times T \dashrightarrow K \times T^*$$

se puede plantear como:

$$\delta : (Q,1,W) = (P1,W1),(P2,W2),\ldots,(Pm,Wm)$$

donde Q, Pi, estan en K, 1 esta en Z, W esta en T y Wi esta en T*, $1 \le i \le m$.

El automatón pushdown en el estado Q0 con el símbolo de entrada 1, y W el símbolo en el tope del almacenamiento pushdown puede para cualquier i, entrar al estado Pi, reemplazar W por Wi, y avanzar la cabeza de entrada un símbolo. La interpretación de $\delta(Q, e, W) = \{(P1, W1), (P2, W2), \ldots, (Pm, Wm)\}$ Es que el automatón pushdown en el estado Q, independientemente del símbolo de entrada que está siendo analizado y con W en el tope del almacenamiento pushdown puede entrar al estado Pi y reemplazar W por Wi para cualquier i, $1 < i < m$, en este caso, la cabeza de entrada no avanza.

Ejemplo 2.11 Automata pushdown.

Un automatón pushdown que acepta $\{WW^R \mid W$ esta en $(0,1)^*\}$ * por almacenamiento vacío sería.

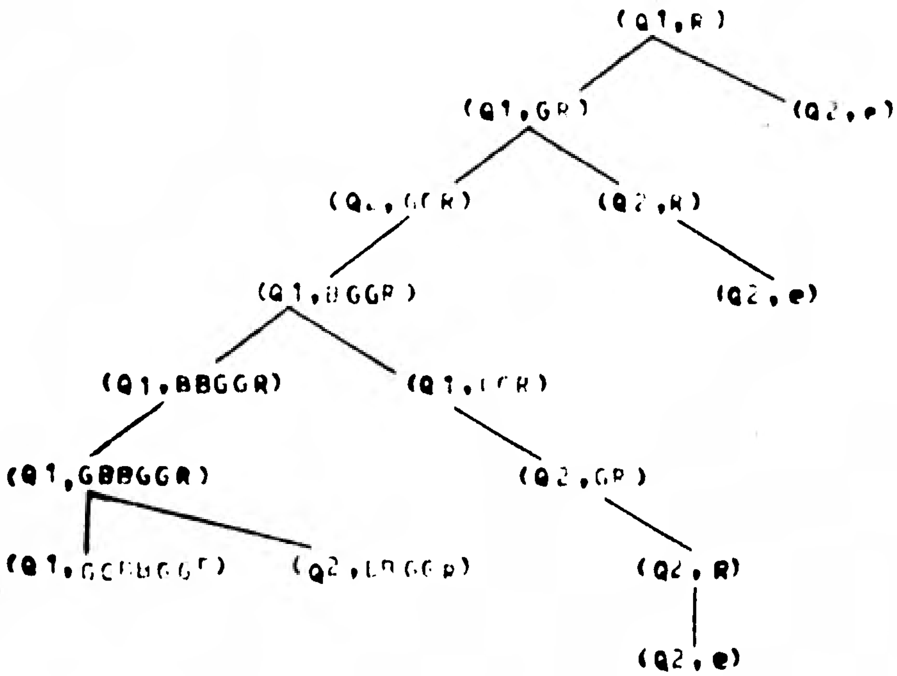
$M = (\{Q1, Q2\}, \{0, 1\}, \{R, B, G\}, \delta, Q1, R, 0)$

cuya función de transición estaría dada por:

1.- $\delta(Q1, 0, R) = \{(Q1, BR)\}$
2.- $\delta(Q1, 1, R) = \{(Q1, GR)\}$
3.- $\delta(Q1, 0, B) = \{(Q1, BB), (Q2, e)\}$
4.- $\delta(Q1, 0, G) = \{(Q1, BG)\}$
5.- $\delta(Q1, 1, B) = \{(Q1, GB)\}$
6.- $\delta(Q1, 1, G) = \{(Q1, GG), (Q2, e)\}$
7.- $\delta(Q2, 0, B) = \{(Q2, e)\}$
8.- $\delta(Q2, 1, G) = \{(Q2, e)\}$
9.- $\delta(Q1, e, R) = \{(Q2, e)\}$
10.- $\delta(Q2, e, R) = \{(Q2, e)\}$

* W^R significa la cadena inversa de W

la configuración de las funciones de transición en el proceso de la cadena 11001' sería:



A continuación planteamos el teorema cuya demostración aparece en el apéndice A.

Teorema 7.- Si L es un lenguaje libre del contexto entonces existe un autómatón pushdown M, tal que L = N(M).

AUTOMATAS LINEALES ACOTADOS

Los automatones lineales acotados son dispositivos que nos permiten reconocer los lenguajes sensibles del contexto y están constituidos por un control finito, una cinta de entrada dividida en cuadros y una cabeza a la cinta que analiza un cuadro cada vez además un automatón lineal acotado es definido como un sistema M = (K,Z,T,δ,Q0,F) donde:

1.- K es un conjunto finito de estados.

2.- T es un conjunto finito de símbolos permitidos en la cinta, uno de los cuales es el símbolo blanco designado por B.

3.- Z ⊆ T conjunto de símbolos de entrada.

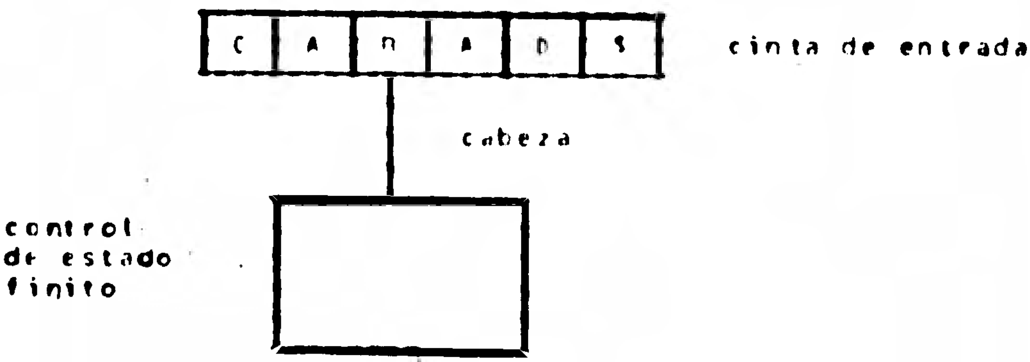
4.- δ Es una función de KxT en K x (T - {¢}) x {I,D})
donde I,D significa que la cabeza a la cinta realiza un movimiento al cuadro izquierdo o al cuadro derecho respectivamente.

5.- Q0 es un estado que pertenece a K y significa el estado inicial.

6.- F contenido en K es el conjunto de estados finales.

Además Z tiene dos símbolos especiales ¢ y $ que significan las marcas finales a la izquierda y a la derecha de la cinta, se encuentran inicialmente a la entrada y al final y tienen por objeto prevenir a la cabeza de lectura la región que está siendo analizada. Fig 2.9

Fig 2.9 Autómata lineal acotado.



Una configuración para un automatón lineal acotado es definida como:

$(Q, A_1, A_2 \ldots A_n, i)$ donde,

Q está en K, $A_1 A_2 \ldots A_n$ es una cadena de símbolos en T, i es un entero entre 1 y n. Además se define una relación como $\vdash_M$ para dos configuraciones donde la segunda es obtenida de la primera mediante la aplicación de una regla de la función de transición δ, es decir.

Si $1 < i < n$ y $\delta(Q, A_i) = (P, A, i)$ entonces

$$(Q, A_1 A_2 \ldots A_n, i) \vdash_M (P, A_1 A_2 \ldots A_{i-1} A A_{i+1} \ldots A_n, i-1)$$

Si $\delta(Q, A_i) = (P, A, D)$ entonces

$(Q,A_1A_2\ldots A_n,i) \vdash (P,A_1A_2\ldots A_{i-1}AA_{i+1}\ldots A_n,i+1)$ esto es

M imprime A sobre A_i, cambia su estado a P, y mueve la cabeza a la izquierda o la derecha. Se define la relación $\vdash^{*}_{M}$

por $(Q,W,i) \vdash^{*}_{M} (Q,W,i)$

si

$(Q_1,W_1,i_1) \vdash (Q_2,W_2,i_2)$ y

$(Q_2,W_2,i_2) \vdash (Q_3,W_3,i_3)$

entonces

$(Q_1,W_1,i_1) \vdash^{*}_{M} (Q_3,W_3,i_3)$.

El lenguaje aceptado por M se define como:

$\{X \mid X$ está en $(\Sigma - \{¢,\$\})^{*}$ y
$(Q_0, ¢X\$, 1) \vdash^{*}_{M} (Q,W,i)$ para alguna
Q en F, y W en Γ^{*}, y un entero $i\}$

A continuación se establece el teorema que relaciona a los lenguajes sensibles de contexto y los autómatas lineales acotados y cuya demostración aparece en el apéndice A.

Teorema 2.3 Si L es un lenguaje sensible al contexto, L es aceptado por un autómata lineal acotado.

2.3 CONJUNTOS BORROSOS

Los conjuntos borrosos fueron desarrollados por Zadeh 1965, para proporcionar un modelo matemático que describa las clases de objetos que no tienen una caracterización exacta dentro de un conjunto, es decir, para aquellas colecciones de objetos que no tienen un criterio de membresia definido con precisión y sea imposible determinar cuando pertenecen o no pertenecen a un conjunto.

Ejemplo 2.17 Conjuntos borrosos.

A) "La clase de animales veloces"

B) "La clase de números reales próximos a cero"

C) "La clase de personas saludables"

Los conjuntos borrosos, es decir, las clases con un grado de membresia continuo, constituyen un aspecto muy importante en áreas como lingüística, autómatas y en general en procesamiento de información.

A continuación se plantean las definiciones básicas.

Definición 2.1 Sea X un conjunto ordinario, un subconjunto borroso A de X está caracterizado por una función de membresía que asocia a cada y de X, un número real en el intervalo [0,1], la función de membresía o característica es denotada por F_A (y) y representa el grado de membresía de y en A es decir, el valor más próximo de F_A (y) a la unidad, constituye un grado mayor de pertenencia de y en A, y entre más próximo este F_A (y) a cero constituye un menor grado de pertenencia de y en A, por lo tanto un subconjunto borroso tiene posibles grados de membresía en un intervalo continuo mientras que un conjunto ordinario solo toma los valores 1 o 0 para cuando y pertenece o no pertenece al conjunto.

Ejemplo 2.13 Caracterización de un conjunto borroso.

Si X es el conjunto de números reales y A es el conjunto borroso de números próximos a cero, entonces, se puede dar una caracterización de A, especificando F_A (y) como una función de X, algunos valores representativos de F_A (y) podrían ser: F_A (2) = 0, F_A (1) = 0, F_A (0.5) = 0.3, F_A (0.4) = 0.8 F_A (0.2) = 1, F_A (0.1) = 1.

Las siguientes definiciones son simples extensiones de algunas definiciones de los conjuntos ordinarios.

Definición 2.2 Un conjunto borroso A es vacío si y solo si es idénticamente cero en X, es decir cuando la función de membresía de A es idénticamente cero.

Definición 2.3 Los conjuntos borrosos A y B son iguales (se denota por $A = B$) si y solo si $F_A(y) = F_B(y)$ para toda y en X (se denota $F_A = F_B$).

Definición 2.4 El complemento de un conjunto borrosos A (se denota como A') se define como $F_{A'}(y) = 1 - F_A(y)$ para toda y en X.

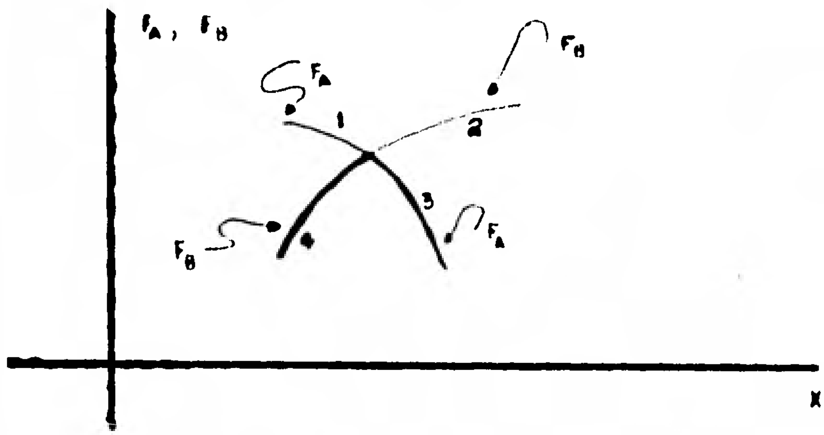
Definición 2.5 El conjunto borroso A esta contenido en el conjunto borroso B (se denota por $A \subseteq B$) si y solo si $F_A(y) \leq F_B(y)$ para toda y en X.

Definición 2.6 La unión de dos conjuntos borrosos A y B (se denota por $A \cup B$) es el conjunto $C = A \cup B$ caracterizado por la función $F_C(y) = max[F_A(y), F_B(y)]$ (se denota por $F_A(y) \vee F_B(y)$) para toda y en X. Max significa el mayor de los dos grados de membresia. Es de notarse que $A \cup B$ es el conjunto borroso mas pequeño que contiene A y B, es decir si D es un conjunto borroso que contiene A y B, entonces para cada y en X, $F_A(y) \leq F_D(y)$ y $F_B(y) \leq F_D(y)$ tal que $F_C(y) = F_A(y) \vee F_B(y) \leq F_D(y)$ de aqui se desprende que $A \cup B = C \subseteq D$.

Definición 2.7 La intersección de dos conjuntos borrosos A y B es el conjunto borroso C (se denota por $A \cap B$) se define como $F_C(y) = min[F_A(y), F_B(y)]$, y e X, se denota por $F_A \wedge F_B$, la intersección de A y B constituye el conjunto borroso mas grande que esta contenido en A y B debido a un argumento dual de la unión de dos conjuntos borrosos.

La interpretación gráfica de la unión e intersección de dos conjuntos borrosos sobre el eje real(R) se ilustra en la fig. 2.10 donde los segmentos de curva 1 y 2 representan la unión y los segmentos 3 y 4 representan la intersección (líneas gruesas).

Fig. 2.10 Unión e intersección de dos conjuntos borrosos en R'.



PROPIEDADES DE LOS CONJUNTOS BORROSOS

Teorema 2.4 Las operaciones de unión, intersección son conmutativas, asociativas y distributivas, es decir si A,B,C son conjuntos borrosos entonces.

1.- $A \cup B = B \cup A$
2.- $A \cap B = B \cap A$
3.- $A \cup (B \cup C) = (A \cup B) \cup C$
4.- $A \cap (B \cap C) = (A \cap B) \cap C$
5.- $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
6.- $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

para probar 6 hacemos $D = A \cup (B \cap C)$. Entonces para cada y en X,

$$
\begin{aligned}
F_D(y) &= F_A(y) \vee (F_B(y) \wedge F_C(y)) \\
&= (F_A(y) \vee F_B(y)) \wedge (F_A(y) \vee F_C(y))
\end{aligned}
$$

tenemos que

$$D = (A \cup B) \cap (A \cup C)$$

es decir

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$

Se puede verificar la identidad anterior considerando los seis casos posibles de $F_A(y)$ y $F_B(y)$ cuando,

$$F_A(y) > F_B(y) > F_C(y), \quad F_A(y) > F_C(y) > F_B(y)$$
$$F_B(y) > F_C(y) > F_A(y), \quad F_C(y) > F_A(y) > F_B(y)$$
$$F_C(y) > F_B(y) > F_A(y), \quad F_B(y) > F_A(y) > F_C(y)$$

para el primer caso tenemos que,

$$\begin{aligned} F_A \vee (F_B \wedge F_C) &= \max [F_A , \min [F_B , F_C]] \\ &= \max [F_A , F_C] \\ &= F_A \end{aligned}$$

y para

$$\begin{aligned} (F_A \vee F_B) \wedge (F_A \vee F_B) &= \min [\max [F_A , F_B], \max [F_A , F_C]] \\ &= \min [F_A , F_A] \\ &= F_A \end{aligned}$$

lo que demuestra la identidad. Los siguientes casos son resueltos en forma semejante.

Teorema 2.5 Las leyes de Demorgan's se cumplen para conjuntos borrosos, es decir

1.- $(A \cap B)' = A' \cup B'$
2.- $(A \cup B)' = A' \cap B'$

Prueba. Para la primera ley, si A y B son conjutos borrosos de X y $y \in X$, entonces si $C = (A \cap B)$

$$F_{C'}(y) = (F_A(y) \wedge F_B(y))' = F_{A'}(y) \vee F_{B'}(y)$$

tal que

$$F_C(y) = F_A(y) \vee F_B(y)$$

es decir

$$(A \cup B)' = A' \cap B'$$

se puede verificar la identidad anterior considerando los dos casos posibles,

a) $F_A > F_B$

b) $F_B > F_A$

para el caso a) tenemos,

$$\begin{aligned}(A \cap B)' &= 1 - \max[F_A, F_B] \\ &= 1 - F_A \\ &= F_{A'}\end{aligned}$$

y

$$\begin{aligned}A' \cap B' &= \min[1 - F_A, 1 - F_B] \\ &= \min[F_{A'}, F_{B'}] \\ &= F_{A'}\end{aligned}$$

para el caso b se demuestra en forma similar.

CLASES DE OPERACIONES EN CONJUNTOS BORROSOS

El álgebra de los conjuntos puede ser extendida formando combinaciones y relaciones entre estos, lo cual amplia más las perspectivas de aplicación y proporciona una estructura teórica mas sólida. Para tal efecto se plantean las siguientes definiciones.

Definición 2.6 El producto algebraico de dos conjuntos borrosos A y B (se denota por AB) se define por la relación de sus funciones de membresia como:

$$f_{AB} = f_A f_B$$

claramente

$$AB \subset A \cap B$$

debido a que

$$A \cap B = \min[f_A , f_B] \quad y$$

$$AB \leq f_A \quad o \quad AB \leq f_B$$

Definición 2.9 La suma algebraica de dos conjuntos borrosos A y B (se denota por A + B) se define por

$$F_{A+B} = F_A + F_B$$

con tal que F + F sea menor que o igual que la unidad, es decir la suma algebraica solo es significativa si $F_A(y) + F_B(y) \leq 1$ para toda y en X.

Definición 2.10 La diferencia absoluta de dos conjuntos borrosos A y B (se denota por I A - B I) se define como:

$$F_{|A-B|} = | F_A - F_B |.$$

Definición 2.11 combinación convexa. Una combinación convexa de dos vectores x,y en K de E y $0 \leq t \leq 1$ se define como:

$$(1 - t)x + ty = x + t(y - x)$$

además si tal combinación esta en K, entonces K es convexo, esta definición puede ser generalizada a conjuntos borrosos de la siguiente forma:

Si A, B y Δ son conjuntos borrosos, la combinación convexa de A,B y Δ (se denota por (A,B;Δ) se define como

$$(A,B;\Delta) = A\Delta + \Delta' B$$

donde, Δ' es el complemento de Δ, la combinación convexa de A, B y Δ en términos de funciones de membresia sería:

$$F_{(A,B;\Delta)}(y) = F_A(y)F_\Delta(y) + [1 - F_\Delta(y)]F_B(y), \quad y \in X.$$

Teorema 2.6 si A y B son conjuntos borrosos, entonces para todo conjunto borroso Δ,

$$A \cap B \subseteq (A,B;\Delta) \subseteq A \cup B$$

prueba. Tenemos que para $0 \leq t \leq 1$ entonces, $t\, F_A(y) + (1 - t)\, F_B(y)$ se encuertra en $[F_A(y), F_B(y)]$ por lo que

$$\min[F_A(y),\ F_B(y)] < t\, F_A(y) + (1-t)\, F_B(y) < \max\, [F_A(y),\ F_B(y)]$$

para y en X.

Además si C satisface $A \cap B \subset C \subset A \cup B$, el conjunto borroso Δ es posible, tal que $C = (A,B;\Delta)$, y la función de membresia de Δ, esta dada por

$$F_\Delta(y) = \frac{F_C(y) - F_B(y)}{F_A(y) - F_B(y)}$$

Definición 2.12 Relación borrosa, la aplicación de relación en conjuntos ordinarios tiene una extensión natural a conjuntos borrosos. Una relacion borrosa en X es un conjunto borroso en el espacio producto X x X .

Ejemplo 2.14 Relación borrosa.

$A = \{ x,y \in R^2 \mid x$ es mucho mayor que $y \}$

puede ser considerado como un conjunto borroso en R, cuya función de membresia $F_A(x,y)$ tentativamente podría tener los siguientes valores: $F_A(20,18) = 0$, $F_A(10,8) = 0.7$, $F_A(50,4) = 1$, etc. Es decir, la relación entre los elementos de dos conjuntos X e Y es una función de membresia $\mu : X \times Y \dashrightarrow [0, 1]$ que asigna a cada par de elementos (x, y), x en X e y en Y, un número $\mu(x, y)$ en el intervalo cerrado $[0, 1]$ indicando el grado en que X esta relacionado con Y.

La relación borrosa binaria puede ser generalizada definiendo la relación borrosa n-aria en X como un conjunto borroso A en el espacio producto $X \times X \times \ldots \times X$, y la función de membresia para funciones de esta naturaleza sería de la forma $F(y_1, \ldots y_n)$, donde $y_i \in X$, $i = 1, \ldots, n$.

En el caso de relaciones borrosas binarias, la composición de dos relaciones borrosas A y B es denotada por $B \cdot A$, y es definida como una relación borrosa en X cuya función de membresia es relativa a los conjuntos borrosos A y B por:

$$F_{A \cdot B}(x,y) = \sup_v \min [F_A(x, v), F_B(v, y)].$$

Definición 2.17 Una relación borrosa S de un conjunto X a un conjunto Y, es un conjunto borrosos del producto cartesiano $X \times Y = \{ (x,y) \mid x \in X \; y \; y \in Y \}$. Es decir, la relación borrosa S es un conjunto de pares ordenados tal que si S es un conjunto borroso de tales pares y el par $(x,y) \in S$, se dice, entonces que "x esta en la relación borrosa S respecto de y" o más especificamente para un par ordenado (x,y), el grado de membresia $\mu_S(x,y)$ de (x,y) en S será definido como la "potencia" de la relación entre x y y.

Ejemplo 2.15 Relación borrosa.

Si X = Y = R = la recta real, entonces << (mucho menor que) es una relación borrosa y constituye un conjunto borroso de pares ordenados (x,y) tal que x,y estan caracterizados por la relación borrosa <<.

El dominio de S es un conjunto borroso en X designado por dom(S) y definido por

$$W_{DOM(S)}(x) = \bigvee_{y} W_S(x,y)$$

donde $\bigvee_{y}$ designa el supremo sobre el conjunto Y

El rango de S es un conjunto borroso en Y designado por ran(S) y definido por

$$W_{RAN(S)}(y) = \bigvee_{x} W(x,y)$$

donde $\bigvee_{x}$ designa el supremo sobre el conjunto X

Definición 2.14 El soporte de un conjunto borroso A de X es un conjunto no-borroso designado por sop(A) y definido como:

$$Sop(A) = \{ x \mid W(x) > 0 \}$$

Definición 2.15 La cardinalidad de un subconjunto borroso A con un soporte finito es designado por I A I y definido por

$$| A | = \sum_i W x_i$$

donde $x_i \in sop(A)$. La cardinalidad de un conjunto borroso es esencialmente una generalización del numero de miembros de un conjunto no borroso.

Definición 2.16 Sea A un conjunto borroso en $X = X_1 \times X_2 \times \dots \times X_n$ y sea $X_i = (x_i)$ Para $i = 1, \dots, n$, con función de membresía $w(x_1, \dots, x_n)$, entonces la "sombra" de A sobre $X_2, \dots, X_n$ es un conjunto borroso en $X_2, \dots, X_n$ cuya función de membresía W_1 esta dada por

$$W_1(x_2, \dots, x_n) = \bigvee_{x_1} W(x_1, \dots, x_n)$$

donde el supremo $\bigvee_{x_1}$ es tomado sobre todas las x_1 en X_1

Definición 2.17 Un conjunto borroso A en X se dice que es normal si

$$\bigvee_{x} W(x) = 1$$

en otro caso se dice que A es subnormal. Si A tiene un soporte finito (es decir sop(a)($W(x)>0$) es finito), entonces A es normal si existe algún x cuyo grado de membresía en A sea la unidad.

Se continua exponiendo brevemente las siguientes propiedades de los conjuntos borrosos.

Definición 2.18 Conjuntos borrosos inducidos por mapeo. Si T es una función de X a Y. Sea B un conjunto borroso en Y con función de membresia $F_B(y)$, la función inversa T^{-1} induce un conjunto borroso A en X cuya función de membresia es definida por $F_A(x) = F_B(y)$ $y \in Y$, para toda x en X que son mapeadas a Y por T.

Definición 2.19 Conjuntos borrosos convexos. Sea X el espacio euclideano E, un conjunto borroso A, es convexo si y solo si los conjuntos

$$Tp = \{ x \mid A(x) \geq p \}$$

son convexos para toda p en (0,1], es decir A es convexo si

$$A[tx1 + (1 - t)x2] > \min [F_A(x1), F_A(x2)]$$

para toda x1, x2 en X y t en [0,1]

Teorema 2.7 Si A y B son conjuntos borrosos convexos su intersección es un conjunto borroso convexo.

Demostración .- Apéndice A.

3.- FUNDAMENTOS TEORICOS DE LOS LENGUAJES BORROSOS

A partir de 1965 en que Zadeh desarrolla la teoría de lenguajes borrosos, se ha ampliado su aplicación a temas de diverso interés, como son: teoría de control, automatas, topología, inteligencia artificial, linguística, etc.

Al final del capítulo 2 se establecen algunas proposiciones de los conjuntos borrosos, que serán utilizados en el desarrollo de los lenguajes borrosos, sin embargo los fundamentos, de los conjuntos borrosos permiten ampliar el campo de operaciones tales como, producto, imagen, imagen inversa, exponenciación, etc., con lo cual se obtienen nuevas aplicaciones de gran utilidad en teoría de lenguajes, fundamentalmente.

Las aplicaciones a la teoría de lenguajes hacen posible obtener resultados en otras áreas como matemáticas o las arriba mencionadas.

La teoría de conjuntos borrosos se puede considerar incluida en la teoría de conjuntos ordinarios con lo cual hereda el rigor matemático usual de estos últimos, sin embargo cualquier apariencia en sentido contrario, es superado definiendo un conjunto borroso como $F_A(x):X \dashrightarrow [0,1]$, $x \in X$, ademas los fundamentos de los conjuntos borrosos al identificar los conceptos de "imprecisión", el conjunto de operaciones aplicado a tales conceptos son determinados en forma natural y puede ser considerada la estructura de los conjuntos borrosos en forma independiente. Otro aspecto interesante de los conjuntos borrosos es el hecho de poder reemplazar el intervalo [0,1] por una estructura más general V, tal como una latiz distributiva.

Los lenguajes juegan un papel fundamental en el desarrollo de las actividades humanas y son estudiados y sistematizados desde diferentes puntos de vista y con diferentes objetivos, sin embargo debido a las imprecisiones y generalidades que se presentan en las palabras o sentencias se puede admitir que para representar un lenguaje natural en general con exactitud, se debe contar con un modelo, como se planteó en el inicio de la sección 2.1 es decir un modelo que relacione esencialmente los símbolos y las ideas, los símbolos con los objetos.

La teoría de lenguajes formales define los lenguajes como un conjunto (finito o infinito) de oraciones, cada una de ellas de una longitud finita y construida a partir de un conjunto de elementos finito. Sin embargo, como se hizo notar más arriba, esta manera de representar un lenguaje no refleja en forma genuina la esencia de un lenguaje natural, el de hacerle corresponder a los objetos cadenas de palabras, o aproximar de manera directa al símbolo con el concepto.

Los lenguajes borrosos en su definición permiten la correspondencia entre cadenas de palabras al conjuntos de objetos, es decir del conjunto de palabras a los conceptos, con lo cual constituye un sistema más completo de representar un lenguaje.

Definición 3.1 Un lenguaje borrosos esta contituido por un sistema de cuatro entidades (U,T,E,N) donde:

U es el universo del discurso que puede estar constituido por cualquier conjunto de objetos como, conjuntos de situaciones, actitudes, acciones, relaciones, acontecimientos, ideas, conceptos, etc. Por ejemplo U puede estar constituido, por el conjunto de lineas rectas, o el conjunto de lineas rectas y sus relaciones entre si, o por el conjunto de cuerpos geométricos, o por el conjunto de libros en una biblioteca que traten sobre las funciones vectoriales.

T es un conjunto borroso (T : U ---> [0,1]) y esta constituido por los nombres de los subconjuntos borrosos de U y se denomina conjunto de términos, es decir, T esta constituidos por cadenas de letras o palabras extraidas de un alfabeto finito y donde cada palabra, esta separada de otra por un símbolo blanco, por ejemplo T puede ser el conjunto de palabras, frases u oraciones del alfabeto español.

E es un conjunto de combinaciones de símbolos de donde son extraídos los términos, es decir, T es un subconjunto borroso de E. Los símbolos de E son extraídos de un conjunto A en un lenguaje formal, A sería el conjunto Vt (conjunto de símbolos terminales) y E sería el conjunto Vt* (que son las cadenas de longitud finita de Vt)

N es una relación borrosa de E a U, y se denomina, relación de nombres. Esta caracterizada por una función de dos variables μN :Sop(T) x U---> [0,1], es decir μN, asocia a cada par ordenado (x,y), x e T, y e U, el grado de membresia, μN(x,y), de (x,y) en N. Entonces μN se puede interpretar como el grado en que x y y estar relacionados.

Ejemplo 7.1 Lenguaje borroso.

Para ilustrar lo anterior se hace uso del ejemplo 2.7 de la seccion 2.2.1, donde A=(a,b,c,+) es decir, tenemos que el conjunto de terminos T, es un subconjunto borroso de E luego, esta caracterizado por una función de membresia, sea WT:E--->[0,1], que ascoia a cada x de T, su grado de membresia WT(x), en T, entonces el grado de membresia de algunas cadenas de T podría ser el siguiente.

WT(+a)=1.0 WT(a+b)=.8
WT(a+)=.1 WT(++c)=.5

Es decir, WT(x) representa el grado de la cadena entre lo no-gramatical y gramaticalmente correcto.

Ejemplo 7.2 Lenguaje borroso.

Sea U el conjunto de salarios de 1000 a 20000, x el término "clase baja" y y=12000 pesos, se puede tener WN("clase-baja",12000) = 0.4 mientras que WN("clase-alta",12000) = 0, donde clase baja y clase alta son las clases económicas distinguidas en los habitantes de una población.

Definición 3.2 Se define un lenguaje estructurado L como un sistema

L = (U,ST,E,SN)

donde:

U es el universo del discurso como se planteó en la definición 3.1 de esta sección.

T se denomina el conjunto de términos y es un conjunto borroso constituido por los nombres de los subconjuntos borrosos de L.

E es un conjunto de combinaciones de símbolos de un alfabeto A, de donde son extraídos los términos, es decir, T está contenido en E.

ST es un conjunto de reglas sintácticas de L, y proporcionan un algoritmo para calcular la función de membresía uL de el conjunto de términos de T.

SN es un conjunto de reglas semánticas de L y proporcionan un algoritmo para calcular la función de membresía uN de el conjunto borroso relación de nombres N.

N es la relación de nombres de naturaleza borrosa y aplica T en U.

Conceptos de Lenguajes Borrosos y Aplicaciones

Definición 3.2 Un lenguaje ya sea estructurado o no, se define como borroso si el conjunto de términos T o la relación N o ambos son de naturaleza borrosa, es decir un lenguaje estructurado es no-borroso si la semántica y la sintaxis que lo constituyen es no-borrosa. Ejemplo de lenguajes estructurados no-borrosos son los lenguajes de programación en el cual el compilador incorpora las reglas para el cálculo de las funciones de membresía para el conjunto de términos T y la relación de nombres, de esta manera por el uso de reglas sintácticas y semánticas el compilador determina cuando una cadena x es un término en T, y calcula UN(x,y), y ∈ U = (términos en lenguaje de máquina) para decidir que instrucción en lenguaje de máquina corresponde a x. Un lenguaje natural está constituido por una sintaxis y una semántica borrosa es decir que las sentencias por ejemplo del español tienen un grado gramatical intermedio entre lo correcto e incorrecto.

[illegible].1 EL SIGNIFICADO DEL SIGNIFICADO

Uno de los puntos débiles de los lenguajes formales ha sido su interés por poner nombres a los elementos linguísticos y deducir la función de estos nombres, los nombres serán dados a las clases cuando estos se hayan establecido y serán extraídos de los términos de la gramática tradicional, pero las clases no serán definidos por estos nombres.

Los términos de un lenguaje natural tienen diferentes matices dentro del campo conceptual en que se utilicen, por ejemplo en español los adjetivos que pertenecen a la serie relacionada con felicidad son: feliz, contento, animado, animoso, alegre, dichoso, etc. Tienen distintos grados de contraste. Es claro que animado y animoso se diferencian menos al compararlos, que con cualesquiera de los otros términos, el que un elemento pueda pertenecer a varias series y contrastar con los elementos de otras series aumenta la complejidad del asunto. Sin embargo es posible tratar este tipo de problemas basándose en al teoría de de conjuntos borrosos, es decir tratar lo borroso del significado caracterizándolo como una semantica borrosa.

Haciendo uso de la definición de lenguaje borroso es posible dar una definición concreta del concepto de significado.

Definición 2.4 Si la función de membresía que caracteriza a la relación de nombres N es

$\mu N: scr(T) \times U$ ---> [0,1],

con $\mu N(x,y)$ representando la relación entre el término x en T y un objeto y en U, entonces el significado de un término x en T es un subconjunto borroso $M(x)$ de U en el cual el grado de membresía de un elemento y en U es dado por $\mu_{M(x)}(y)=\mu N(x,y)$, es decir $M(x)$ es un subconjunto borroso de U, el cual esta condicionado por x como un parámetro que es una sección de N en el sentido en que su función de membresía $\mu_{M(x)}$:U ---> [0,1] es obtenida asignando un valor particular, x, al primer argumento en la función de membresía N.

Ejemplo 7.- Lenguaje borroso.

Sea un lenguaje $L=(U,T,E,N)$ en donde T es el conjunto de clases económicas baja, media y alta, U es el conjunto de salarios de 1000 a 20000 y N es una relación de nombres de E a U definida subjetivamente por:

$$\mu N(\text{clase-baja},y)=1 \quad \text{para } y < 5000$$
$$=[1+((y-5000)/1000)]^{-1} \quad \text{para } y \geq 5000$$

$$\mu N(\text{clase-alta},y)=0 \quad \text{para } y < 15000$$
$$=[1+((y-15000)/1000)]^{-1} \quad \text{para } y \geq 15000$$

$$\mu N(\text{clase-media},y)=0 \quad \text{para } 0 \leq y \leq 5000$$
$$=[1+((y-10000)/1000)^4]^{-1} \quad \text{para } 5000 \leq y \leq 12000$$
$$=[1+((y-12000)/1000)^2]^{-1} \quad \text{para } y \geq 12000$$

para $1000 \leq y \leq 20000$

Entonces el significado del término clase baja es un subconjunto borroso $M(\text{clase-baja})$ de $U=[1000,20000]$ cuya función de membresía viene dada por

$$\mu_{M(\text{clase-baja})}(y)=1 \quad \text{para } y < 5000$$
$$=[1+((y-5000)/1000)]^{-1} \quad \text{para } y \geq 5000$$

el mismo caso ocurre para clase media y clase alta.

Si N es una relación borrosa de T a U, entonces el dominio de N, D(N) es un conjunto borroso en T que es la imagen "sombreada" de N en T, la función de membresía de D(N) está dada por

$$\mu_{D(N)}(x) = \bigvee_{y} \mu_N(x,y)$$

donde el supremo $\bigvee_{y}$ es tomado sobre toda y en U.

El grado de membresía del término x en D(N) se interpreta como el "grado de significación" de x. De esta manera x es totalmente significativo si M(x) es un conjunto borroso normal, esto es

$$\mu_{D(N)}(x) = \bigvee_{y} \mu_N(x,y) = 1$$

Es decir x es totalmente significativo si existe y tal que $\mu_N(x,y)=1$. Inversamente x, es poco significativo si M(x) es un conjunto vacío, esto es $\mu_N(x,y)=0$ para toda y en U.

Ejemplo [illegible] significado de sentencias borrosas.

Sea U un conjunto de enteros U={1,2,....,10} y los términos frío, caliente, no-frío, no-caliente como el siguiente conjunto borroso de U:

$$M(\text{frío})=\{(1,1.0),(2,1.0),(3,0.8),(4,0.2)\}$$

$$M(\text{caliente})=\{(7,0.2),(8,0.8),(9,1.0),(10,1.0)\}$$

$$M(\text{no-frío y no caliente})=M''(\text{frío}) \cap M''(\text{caliente})$$

$$M(\text{frío y caliente})=M(\text{frío}) \cap M(\text{caliente})$$

donde M" significa el complemento de M y $\cap$ es el símbolo de intersección, lo cual constituye.

$$M(\text{no-frío y no-caliente})=\{(3,0.2),(4,0.8),(5,1.0),(6,1.0),(7,0.8),(8,0.2)\}$$

y

M(frío y caliente)= al conjunto vacío.

De esta manera, no-frío y no-caliente es totalmente significativo mientras que frío y caliente es poco significativo.

Como ha sido expresado en los capítulos anteriores, es muy importante para el significado el contexto en el cual se lleve a cabo el concepto, es decir, para un término x cuando es acompañado de otros componentes depende del contexto en que x ocurra, para obtener el significado. En el ejemplo 3.3 se precisaron el significado de las clases económicas, baja, media y alta, sin embargo es claro que tales clases económicas varían para los empleados del campo y los de la ciudad, una posibilidad de tratar este tipo de situaciones es incorporar a tales términos el significado en base a porcentajes, por ejemplo el término "veloz" se le puede aplicar a un objeto y en un subconjunto Z de U, designando tal velocidad en Z como h(y). De esta manera h50 significa la media de h(y) sobre Z y h, es decir significa el r-porcentaje de h(y) sobre Z, donde un valor de h tal que el porcentaje 100-r de un numero de miembros de Z tiene mayor velocidad que o igual que hr entonces se puede asignar el grado de membresía 0.5 al objeto cuya velocidad es h en el conjunto borroso denominado Z, y el grado de membresía

$\mu_{VELOZ\,Z}(y) = r/100$

a un objeto y cuya velocidad sea h, en general, el grado de membresía de un objeto cuya característica sea h es relativa a r, y tiende a cero o uno respectivamente como r tienda a cero o cien. Lo anterior se puede ilustrar considerando a U como un conjunto de animales, y Z el subconjunto de caballos. Si la velocidad de un caballo es de 20 mts./Seg. Y representa el 50 por ciento de la velocidad en Z. Entonces el grado de membresía de tal caballo en la clase de "caballos veloces" en Z puede ser 0.60. Sin embargo cuando un término x1 tiene dos posibles significados M1(x1) y M2(x2), entonces a x1 se le puede asignar el significado, que maximice el grado de significación en el contexto en que x1 ocurra.

SINTAXIS DE LOS LENGUAJES BORROSOS

Para el desarrollo de la sintáxis de los lenguajes borrosos, son considerados y ampliados los razonamientos expuestos por la teoría de los lenguajes formales.

En el capítulo anterior se definió una gramática formal G=(Vn,Vt,S,P) donde: Vn es el conjunto de símbolos no-terminales y que constituye a las categorías de la gramática, es decir solo tienen significado en cuanto a miembros de un sistema y cada miembro puede ser repartido dentro de las clases siguientes oración, cláusula, grupo, palabra, morfema;Vt a su vez constituye el conjunto de simbolos terminales en español serian las letras de alfabeto,números y un símbolo blanco. Al aplicar la operación cerradura Kleene al conjunto Vt, obtenemos Vt* que sería el conjunto de cadenas x formadas con los simbolos de Vt. Mediante un razonamiento paralelo son definidos los lenguajes borrosos como L = (U,St,Vt*,Sn) donde el conjunto de terminos T, se considera como un conjunto borroso de Vt caracterizado por la función de membresia

$$MT:Vt* ---> [0,1],$$

que asocia, a cada cadena x en Vt* su grado de membresia MT(x), en T.

El conjunto de reglas sintácticas SI de L proporciona las reglas para la generación de cadenas en el sop(T) junto con sus grados de membresía, y constituyen la gramática borrosa de L.

Una forma específica de obtener la gramática borrosa de L es mediante la noción de gramática formal G=(Vn,Vt,P,S) haciendo las pertinentes generalizaciones para los lenguajes borrosos como sigue: S e Vn, es el símbolo inicial P es el conjunto de producciones de la forma

$$u \xrightarrow{\mu} v$$

donde u,v son cadena compuestas de elementos de Vt U Vn, y (u ≠ e) y 0 ≤ μ ≤ 1. Si u ---> v y w1, w2 son cadenas arbitrarias en (Vt U Vn)*, entonces

$$w1uw2 \xrightarrow{p} w1vw2$$

donde se dice que w1vw2 es derivable directamente w1uw2.

Si u,v son cadenas en (Vt U Vn)* y existen cadenas w1,w2,..., wn-1 en (Vt + Vn) tal que

$$u \xrightarrow{p1} w1 \xrightarrow{p2} w2 \xrightarrow{p3} \ldots \xrightarrow{pn-1} wn-1 \xrightarrow{pn} v$$

entonces, se dice que v es derivable desde u a travez de las reglas de derivacion (u,w1,..., wn-1,v). La potencia de (u,w1,..., wn-1,v) se define como el min(p1,..., pn) que operacionalmente es p1 $\wedge$ p2 $\wedge$ ... $\wedge$ pn, la potencia de p de la relación entre u y v se define como:

$$P = \sup \min(p1 \ldots pn)$$

donde el sup se toma sobre todas la reglas de derivación de u a v.

Si w1,...., wm son cadenas en (Vt U Vn) y

$$w1 \xrightarrow{r1} w2 \xrightarrow{p2} \ldots \xrightarrow{pm} wm$$

tal que p2 ,...., pm > 0, entonces se dice que w1 deriva wm en la gramática G, esto es abreviado mediante $w1 \overset{}{\underset{G}{\Longrightarrow}} wm$.

Definición 3.5 Una gramática borrosa G genera un conjunto borroso de términos designado por T(G), como sigue: Una cadena terminal x en T(G), esto es en el soporte de T(G) ($\mu_T(x)>0$) si tal cadena terminal x es derivable desde S (símbolo inicial), el grado de membresia $\mu_T(x)$ es la potencia de la relación entre S y x.

Ejemplo 3.7 Gramática borrosa.

Sea la gramática G constituida por G=(Vn={S,T,F}, Vt={a,(,),+,*}, P,S) donde P consiste de las producciones siguientes:

$$S \xrightarrow{0.4} S + T$$

$$S \xrightarrow{0.6} T$$

$$T \xrightarrow{0.3} T * F$$

$$T \xrightarrow{0.6} F$$

$$F \xrightarrow{0.2} (S)$$

$$F \xrightarrow{0.9} a$$

se considera la cadena terminal x = a + a, las posibles reglas de derivación de tal cadena son

```
          0.4                0.4                 0.4
a)  S   ---> S+T    b)  S  ---> S+T     c)  S  ---> S+T

          0.9                0.8                 0.6
        ---> S+F           ---> T+T            ---> S+F

          0.8                0.8                 0.9
        ---> T+F           ---> F+T            ---> S+a

          0.9                0.5                 0.6
        ---> T+a           ---> a+T            ---> T+a

          0.8                0.7                 0.8
        ---> F+a           ---> a+F            ---> F+a

          0.9                0.5                 0.9
        ---> A+a           ---> a+a            ---> a+a
```

luego

```
WT(a+a)=max(min(0.4,0.8,0.8,0.7,0.8,0.5)
                Min(0.4,0.8,0.8,0.9,0.6,0.9)
                Min(0.4,0.8,0.9,0.8,0.6,0.9)
WT(a+a)=max(0.4,0.4,0.4)
WT(a+a)=0.4
```

3.2 TIPOS DE GRAMATICAS

La clasificación de las gramáticas desarrolladas por Noam Chomsky para los lenguajes formales son extendidas en forma paralela a las gramáticas de los lenguajes borrosos, donde esencialmente las diferencias se presentan en las producciones que las constituyen.

GRAMATICAS TIPO 3 O REGULAR

Las producciones que contituyen este tipo de gramáticas estan caracterizadas por reglas de la siguiente forma:

$$A \xrightarrow{p} aB$$

$$A \xrightarrow{p} Ba$$

$$A \xrightarrow{p} a$$

$$S \xrightarrow{p} e$$

donde $p \in [0,1]$, $a \in V_t$, $A, B, S \in V_n$, e es la cadena vacía.

GRAMATICAS TIPO 2 O LIBRES DEL CONTEXTO

Las producciones estan caracterizadas por las siguientes reglas.

$$A \xrightarrow{p} w$$

$$S \xrightarrow{p} e$$

donde $A, S \in V_n$, $w \in (V_n \cup V_t)^*$, $p \in [0,1]$ tales producciones, como en las gramáticas formales, A puede ser reemplazada por w prescindiendo del contexto en que A aparezca.

GRAMATICAS TIPO 1 O SENSIBLES DEL CONTEXTO.

Estan caracterizadas por producciones de la forma:

$$uAv \xrightarrow{p} uBv$$

$$S \xrightarrow{p} e$$

donde $p \in [0,1]$, u,v y B pertenecen a $(Vn \cup Vt)^*$, $A \in Vn$ y $B \neq e$, además $long(B) \geq long(A)$.

GRAMATICAS TIPO 0

Este tipo de gramáticas estan constituidas por producciones en la expresión más general, a saber.

$$u \xrightarrow{p} v$$

donde u,v son cadenas en $(Vn \cup Vt)^*$, con $u \neq e$.

1.7 SEMANTICA DE LOS LENGUAJES BORROSOS

En los lenguajes naturales, el estudio del significado (semántica), tiene principal interés en el "contenido" de los elementos de T o léxicos que lo constituyen, tales elementos léxicos, están organizados en series que caen dentro de ciertos límites conceptuales. El significado de los elementos léxicos dentro de una serie tal como: Animado, animoso feliz, contento, agradable;depende principalmente de la zona que no comparta con otros elementos, es decir, el significado de un elemento lexico vendría dado por la zona que no tenga en común con otros elementos de la serie léxica de la lengua.

El problema fundamental de la semántica consiste en el hecho de que un elemento léxico pueda pertenecer a varias series y su significado constrastar con los miembros de tales series; la semántica de los lenguajes borrosos por lo tanto tiene como objetivo, especificar el conjunto de reglas semánticas, que constituyan un proceso algorítmico para calcular el significado de sus componentes como se muestra a continuación:

Sea L = (U,ST,E,SN) un lenguaje borroso estructurado tal que, SN, el conjunto de las reglas semánticas son deducidas como se plantea en la sección 3.1, es decir calculando la función de membresía.

$$WN: Sop(L) \times U \rightarrow [0,1]$$

de la relación de nombres N. Específicamente, las reglas semánticas deben ser deducidas de un conjunto finito de pares ordenados

$$\{[(xi,yi), WN(xi,yi)]\} \quad i,j = 1,\ldots,k$$

donde xi,yi son elementos en T y U respectivamente y WN(xi,yi) es la potencia de la relación entre xi y yi, es decir, a un término xi en T, le corresponde el significado yi en U.

Lo anterior se puede ilustrar mediante el siguiente ejemplo.

Ejemplo 3.6 Semántica borrosa.

Se considera el conjunto de términos T=(rápida,lenta) y el universo del discurso compuesto por el conjunto 1 a 1000 tornillos, si se desea saber la rapidez o lentitud de una máquina que fabrica tornillos, se define la relación de nombres N de E a U como sigue:

$$WN(lenta,y) = 1 \quad \text{para } y < 250$$
$$= [1 + ((y-250)/50)^2]^{-1} \quad \text{para } y \geq 250$$

$$WN(rápida,y) = 0 \quad \text{para } y < 500$$
$$= [1 + ((y-500)/25)^2]^{-1} \quad \text{para } y \geq 500$$

utilizando la definición 3.4, que establece lo siguiente: Para cada término $x0$ en T su significado $M(x0)$, que es un subconjunto borroso en U se define por $W_{M(x0)}(y) = W_L(x0,y)$. Por lo tanto el significado de un término compuesto $x1$ o $x2$ es definido por la unión de los argumentos

$$M(x1 \text{ o } x2) = M(x1) \cup M(x2)$$

que en términos de sus funciones de membresía es

$$WN(x1 \text{ o } x2) = WN(x1,y) \vee WN(x2,y)$$

en forma similar para el conectivo "y" se tendría la intersección de los argumentos $x1, x2$ como sigue:

$$M(x1 \text{ y } x2) = M(x1) \cap M(x2)$$

que en términos de su función de membresía se tiene

$$WN(x1 \text{ y } x2, y) = WN(x1,y) \wedge WN(x2,y)$$

es decir, si en una frase $x1x2$, $x2$ representa el adjetivo y $x1$ el nombre, entonces $M(x1)$ y $M(x2)$ son los subconjuntos borrosos de U que representan los significados de $x1$ y $x2$ respectivamente, el significado de la frase $x1x2$ es definida como la intersección de $M(x1)$ y $M(x2)$ tal que

$$M(x1x2) = M(x1) \cap M(x2)$$

Ejemplo 2.- Semántica borrosa.

Sea U un conjunto de edificios en una población y sea x1=casa y x2 = azul, entonces M(x1) es el conjunto borroso de casas en la población y M(x2) es un conjunto borroso de edificios azules, entonces M(x1x2) es el subconjunto borroso de casas azules en la población, es decir si el grado de membresía del conjunto de edificios azules es W(x2) = 0.8 y el grado de membresía del conjunto de casas es W(x1) = 0.4, entonces

$$
\begin{aligned}
M(x1x2) &= M(x1) \cap M(x2)\\
&= W(x1) \wedge W(x2)\\
&= 0.8 \wedge 0.4\\
&= 0.4
\end{aligned}
$$

Ejemplo 3.- Semántica y sintáxis borrosas.

Un ejemplo que ilustra mas ampliamente la noción de sintáxis y semántica de los lenguajes borrosos es el siguiente.

Sea L = (U,ST,E,SM) un lenguaje borroso estructurado donde (rápida,lenta,muy,no,y,o) es el conjunto de términos T, y el conjunto de reglas sintácticas St constituyen las reglas de producción dadas por

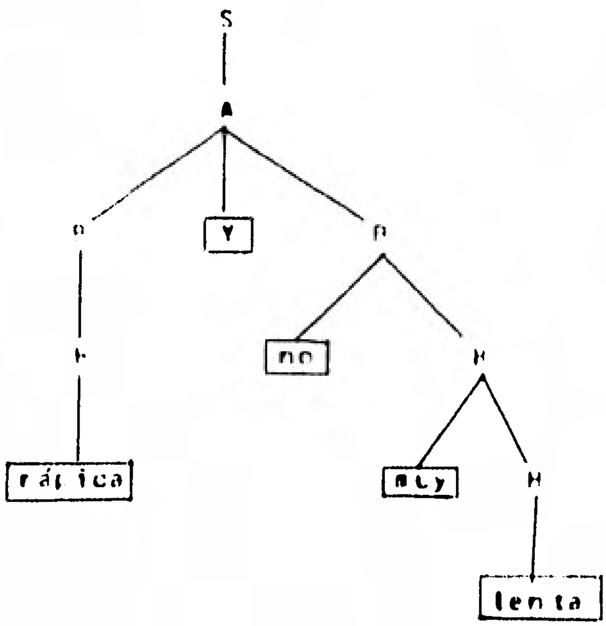
S ——> A
S ——> S o S
A ——> B
A ——> B y B
B ——> H
B ——> no B
H ——> (S)
H ——> rápida
H ——> lenta
H ——> muy H

donde "(", ")" son delimitadores

Las series de términos mas interesantes generados por tales reglas sintácticas son:

Rápida.
Muy rápida.
Rápida y no muy lenta.
Lenta o (no muy rápida).
No muy lenta.
No muy rápida y no muy lenta.

Por ejemplo el árbol de análisis de sintaxis de x = rápida y no muy lenta es:



Para calcular el significado µN(x,y) de un término compuesto x tal como "muy rápida", suponemos que µN(rápida,y) y µN(lenta,y) han sido previamente establecidos (ejemplo 3.6) y se considera µ(E) como una abreviación de µN(E,y) donde E son los símbolos terminales o noterminales, entonces, se pueden expresar las reglas de producción anteriores mediante las funciones de membresía respectivas como:

S ——> A	===> µ(S) = µ(A)
S ——> S o S	===> µ(S) = µ(S) V µ(S)
A ——> H	===> µ(A) = µ(H)
A ——> B y B	===> µ(A) = µ(B) ∧ µ(B)
B ——> H	===> µ(B) = µ(H)
B ——> no H	===> µ(B) = 1 - µ(H)
H ——> (S)	===> µ(H) = µ(S)
H ——> rápida	===> µ(H) = µ(rápida)
H ——> lenta	===> µ(H) = µ(lenta)
H ——> muy H	===> µ(H) = $(µ(H))^2$

entonces, para un término compuesto tal como

x = rápida y no muy lenta

la función de membresía para $M(x)$ se escribe de la siguiente forma:

$$MN(x,y) = MN(\text{rápida}) \wedge (1 - MN(\text{lenta}))$$

la cual se evalúa según la definición de las M del ejemplo 3.6.

4.- APLICACION DE LOS LENGUAJES BORROSOS

El objetivo fundamental de los lenguajes borrosos es obtener un modelo matemático que proporcione una mejor representación de los lenguajes naturales, para finalmente mediante el uso de la tecnología relativa a la computación y el empleo de la inteligencia artificial, aplicar este modelo en las diferentes áreas de las actividades humanas, la implementación de los lenguajes borrosos que se establece a continuación comprende esencialmente de dos partes, el modelo de "aprendizaje", el cual consiste en una técnica que intensifique el grado de membresia de las producciones deseadas y el modelo de "enseñanza" enfocado a una máquina, mediante un conjunto de sentencias secuenciales con el objeto de reforzar el grado borroso de membresia de las producciones de una gramática borrosa inherente a la máquina.

El "aprendizaje" y la "enseñanaza" son algunas de las actividades más intrínsecas de la inteligencia humana debido a la capacidad para manejar información de carácter imprecisa o borrosa, sin embargo se permite aplicar tales términos a las máquinas, en cuanto se refieran a la clasificación de grupos de características de objetos, de acuerdo a ciertas categorias o a la división de categorias en cuanto sean clasificadas.

Una operación muy útil en los modelos de los sistemas de aprendizaje ha sido el proceso de un número muy alto de iteraciones ensayo-y-error que inducen a un refinamiento y formalización concerniente a las relaciones en una gramática generativa a partir de definiciones "primitivas" (producciones) derivadas de un lenguaje formal. La ventaja de utilizar los lenguajes borrosos consiste en evitar la gran cantidad de iteraciones, mediante el uso de grados de membresia.

I MODELO DE APRENDIZAJE

Sea L1 = (U,ST1,E1,SN1) un lenguaje borroso estructurado con el cual se inicia el proceso de aprendizaje de una máquina, donde:

U es el universo del discurso

T es el conjunto borroso constituido por los nombres de los subconjuntos borrosos de U.

E es un conjunto de combinaciones de símbolos de un alfabeto A de donde son extraídos los términos.

ST es un conjunto de reglas sintácticas de L y proporcionan un algoritmo para calcular, la función de membresia WL, del conjunto de términos de T.

SN es un conjunto de reglas semánticas de L y proporcionan un algoritmo para calcular la la función de membresia WN de el conjunto borroso relación de nombres N.

N es la relación de nombres de naturaleza borrosa y aplica E en U.

Las reglas semánticas (SN1) y sintácticas (ST1) se eligen previamente para cubrir un rango suficiente en el lenguaje y para dar un adecuado enfoque del tema. La máquina avanza en el proceso de aprendizaje actualizando en forma secuencial los grados de membresia (semánticos) en SN1 a travez de la aplicación de las reglas sintácticas (producciones) de ST1 más utilizadas, además si el conjunto de reglas sintácticas contenidas en ST1 son R, entonces R puede contener reglas de producción gramaticalmente incorrectas o impropias, tales reglas serán eliminadas o degradadas por el aprendizaje.

En el tiempo n, la máquina tendrá un lenguaje borroso

$$L_n = (U, ST_n, E_n, SN_n)$$

donde

$$SN_n = \{ W_n(u \longrightarrow v) \mid (u \longrightarrow v) \in ST_n \}$$

son las actualizaciones secuenciales de los grados de membresia semánticos.

El aprendizaje de la máquina se inicia con el conjunto finito:

$$K_n = \{ x_{ni} \mid i = 1,\ldots\ldots,h_n \}$$

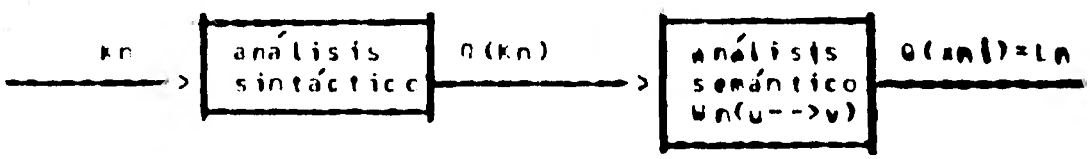
de sentencias, donde $x_{ni} \in L_1$. El análisis de la máquina con respecto a cada x_{ni} constituye a la sucesión del aprendizaje y es posible debido a la recursividad de L_1. Sea $Q(x_{ni})$ el conjunto de todas las reglas de producción (semánticas) que tienen alguna posibilidad de ser usadas para generar cualesquiera x_{ni} y $Q(K_n)$ el conjunto de reglas de producción (sintácticas) para generar K_n, entonces tenemos que $Q(K_n) \subset R$ y $Q(K_n) = \bigcup_{i=1}^{h_n} Q(x_{ni})$. Además la función característica del conjunto $Q(K_n)$ es θ_n, donde, $\theta_n(r) = 1$ para $r \in Q(K_n)$ y $\theta_n(r) = 0$, para $r \notin Q(K_n)$.

Considerando que los grados de membresía de las reglas de producción semánticas (W_n) son más altas, en tanto sean más utilizadas y más bajas en cuanto menos se apliquen, se recurre por simplicidad a la siguiente forma de aprendizaje lineal.

$$W_{n+1}(u \dashrightarrow v) = sW_n(u \dashrightarrow v) + (1-s)\theta_n(u \dashrightarrow v), \quad 0 < s < 1. \quad (1)$$

El modelo de aprendizaje se puede representar mediante el diagrama siguiente:

Fig.4.1 Modelo de aprendizaje de un lenguaje borroso.



Cuando una sentencia es ambigua,es decir tiene más de dos reglas de derivación,se consideran los siguientes métodos de aprendizaje.

Método 1.-Se intensifican todas las reglas de producción que pueden ser usadas en alguna derivación con el objeto de elevar el grado de membresia.

Método 2.-Cuando la derivación correcta es directa, unicamente la regla de producción correcta es intensificada.

Ambos métodos son similares en el caso en el cual las sentencias sean derivadas sin ambiguedad,por lo tanto unicamente será considerado el método 1,es decir si la sentencia es ambigua se puede intensificar unicamente las reglas especificadas eligiendo una adecuada sucesión de enseñanza.

2.- MODELO DE ENSEÑANZA

Si

$$L_{nj}=\{x \mid \mu_n(S \Longrightarrow x) \geq j\}$$

Entonces L_{nj} es el conjunto de todas las sentencias x que son derivadas a partir del símbolo inicial de las reglas de producción sintácticas y cuyo valor semántico es igual que o mayor que j, donde j y $\{K_i \mid i=1,2,\ldots,n-1\}$ se deben elegir adecuadamente con la finalidad de tener un conjunto deseado de sentencias (L_{nj}) cerrado. Además, se asume

$$R_{nj}=\{r \mid r \in R,\ \mu_n(r) \geq j\}$$

como el conjunto de reglas de producción semánticas que tienen la posibilidad de ser usadas para generar las sentencias x de L_{nj}, es decir:

$$R_{nj} = L_{nj} \qquad (2)$$

por lo tanto, si la máquina da únicamente sentencias gramaticalmente correctas, R_{nj} contendrá únicamente reglas de producción gramaticalmente correctas.

Teorema 4.1 Si para toda n, $R_n \subseteq K$, entonces para toda $j \in (0,1)$, existe $N(j)$ tal que, para toda $n \geq N(j)$, $L_{jn} = L(G(K))$. Además

A) $\forall$ n, $L_n = L_{01} = L(R)$
B) $\forall$ j,n $L_{jn} = L_{01}$
C) $K \subseteq L\, G(K) \subseteq L_1$

para probar la primera inferencia se considera la ecuación (1) donde para cualesquiera regla de producción $r \in G(K)$ con grado semántico inicial $W_1(r)$ y para cuando tal regla es aplicada n veces, obtenemos el grado semántico $W_{n+1}(r)$ mediante la siguiente ecuación

$$W_{n+1}(r) = s^n W_1(r) + (1-s)\sum_{i=0}^{n-1} s^i \theta_{n-i}(r)$$

Existe que $r \in G(K)$, entonces $\theta_n(r) = 1$ para $n = 1,2,...$ puesto que su grado de membresía es total y como

$$\sum_{i=0}^{n-1} s^i = 1 - s^n / 1 - s$$

$$W_{n+1}(r) = s^n W_1(r) + 1 - s^n$$

por lo tanto, para toda j existe $N'(j)$, para toda $n \geq N'(j)$, se tiene $W_n(r) \geq j$, es decir $\lim_{n \to \infty} W_n(r) = 1$ ya que $s \dashrightarrow 0$ cuando $n \dashrightarrow \infty$. En forma similar, si $r \notin G(K)$, entonces para toda j, existe $N''(j)$, para toda $n \geq N''(j)$, se tiene $W_n(r) \leq j$, es decir $\lim_{n \to \infty} W_n(r) = 0$. Sea $N(j) = \max[N'(j), N''(j)]$ entonces, para toda $n \geq N(j)$, tenemos que $R_{nj} = G(K)$ usando la igualdad (?), la primera inferencia del teorema queda probada.

Prueba: A) se debe al hecho de que $\forall n, r \in R$, $Wn(r) \geq 0$. Es decir

$LCn = \{ x \mid Wn(x) \geq 0 \}$
$LC1 = \{ x \mid W1(x) \geq 0 \}$
$L(R) = \{ r \mid Wn(r) \geq 0 \}$

donde L(R) es el lenguaje generado por las reglas de producción R.

B) la prueba es obvia debido a que $Lin \subseteq L01$

C) la prueba de este inciso se desprende a partir de la definición de Q(K)

Ejemplo 4.1 Aprendizaje de un lenguaje borroso.

Sean las reglas de producción sintácticas las siguientes

S ---> c
S ---> aSb
S ---> abS

cuya expresión semántica está dada por:

W1(S ---> c) = 0.5
W1(S ---> aSb) = 0.5
W1(S ---> abS) = 0.5

Sea S = 0.6 y $r = (a^2 c b^2)$ el proceso de aprendizaje se muestra en la tabla 1.1, mientras que

$Q(K) = \{ S \text{ ---> } c,\ S \text{ ---> } aSb \}$

$$L_{0.7^N} = \begin{cases} \emptyset & N = 1,2,3 \\ \{a^k c\, b^k \mid k = 0,1,2,\ldots\} = L(Q(K)), & n = 4,5,\ldots \end{cases}$$

$L(R) = \{ a^i (ab)^j c\, b^i,\ (ab)^k a^i (ab)^j c\, b^i, \ldots \mid i,j,k = 0,1,2,\ldots$
$\supseteq L_n,\ n = 1,2,\ldots$

W	W1	W2	W3	W4 ...
S ---> c	0.5	0.6	0.68	0.744 ↗
S ---> aSb	0.5	0.6	0.68	0.744 ↗
S ---> abS	0.5	0.4	0.32	0.256 ↘

tabla 1.1 Proceso de aprendizaje, las flechas indican cuando las producciones más usadas son crecientes.

CONCLUSION

Las gramáticas borrosas son formuladas usando los conceptos de conjuntos borrosos definidos por Zadeh con lo cual constituyen un método de contrucción de un clasificador determinístico de cadenas o una gramática sin ambiguedad y su estudio esta enfocado al desarrollo de modelos en los cuales se precise la existencia de indeterminación, duda o vaguedad en vez de aleatoriedad.

La teoría de lenguajes borrosos constituyen la continuación del modelo matemático para los lenguajes naturales planteado por Noam Chomsky en los lenguajes formales y establecen el punto de partida hacia una mejor precisión de las nociones de sintáxis y semántica, conceptos fundamentales en linguística.

La contribución de la teoría de los lenguajes borrosos ha permitido reducir las diferencias entre la precisión de los lenguajes formales y la imprecisión que caracteriza las expresiones en los lenguajes naturales y contribuye a un mejor conocimiento de aquellos procesos que involucran la manipulación de datos borrosos.

La teoría de lenguajes borrosos tiene un campo de aplicación muy interesante en el diseño de lenguajes de programación por ejemplo, el algol 60 como se ha dicho fue desarrollado por John Backus mediante la notación "forma normal de Backus" FNB (apéndice b) y que equivalen a las gramáticas libres del contexto de los lenguajes formales definidos por Noam Chomsky, sin embargo las limitaciones más importantes de este lenguaje son su incapacidad en el manejo de clases de objetos que no tienen un grado de membresia exacto dentro de un conjunto, una solución especifica para este problema en particular la proporciona la implementación de las gramáticas libres del contexto borrosas y en general mediante el uso de las gramáticas borrosas.

APENDICE A

Teorema 2.1 Sea G =(Vn,Vt,P,S) una gramática tipo 3 entonces existe un automatón finito M =(K,Vt,δ,S,F) con T(M) = L(G).

prueba: Si X=a1,a2,...., an esta en L(G), n ≥ 1. Entonces

S ⇒ a1A1 ===>...===> a2A2... an-1An-1 ===> a1... an-1an

para alguna sucesión de variables A1,A2,....,AN-1. De la definición de δ se puede ver que δ(S,a1) contiene A1, que δ(A1,a2) contiene A2,etc, y que δ(An-1,an) contiene A (estado final). Así X esta en T(M), puesto que δ(S,X) contiene A, y A esta en F (estados finales). Si e esta en L(G) entonces S esta en F así e esta en T(M), luego L(G) ⊂ T(M).

En otro caso, si X esta en T(M), long(X) ≥ 1, entonces existe una sucesión de estados S,A1,A2,...,An-1,A tal que δ(S,a1) contiene A1, δ(S,a2) contiene A2, y así hasta el final. Así, P contiene reglas S ---> a1A1, A1 ---> a2A2,.... y An-1 ---> an. Así, S ===> a1A1 ===> a1a2A2 ===>...===> a1a2... an-1An ===> a1a2... an es una derivacion en G y X esta en L(G). Si e esta en T(M), entonces S esta en F, así S ---> e es una producción en P y e esta en L(G) luego T(M) ⊂ L(G).

Teorema 2.7 Si L es un lenguaje libre del contexto, entonces existe un autómata pushdown M, tal que L = N(M)

Prueba: Sea G = (Vn,Vt,P,S) una gramática libre del contexto, sea

$$M = (\{ Q1 \}, Vt, Vn, \delta, Q1, S, \emptyset)$$

donde $\delta(Q,a,A)$ contiene (Q,Z) siempre que $A \rightarrow aZ$ este en P. Para mostrar que L(G) = N(M), se considera lo siguiente:

$$xAB \underset{G}{\Longrightarrow} xaUB \text{ si y solo si } a: (Q1,AB) \vdash_{M} (Q1,UB).$$

Por inducción sobre el número de pasos de la derivación que

$xAB \overset{*}{\underset{G}{\Longrightarrow}} xya$, para cualesquiera x y y en Vt*,

$a \in Vn$, U y B en Vn* si y solo si

$$y: (Q,AB) \vdash^{*}_{M} (Q,a).$$

Así $S \overset{*}{\underset{G}{\Longrightarrow}} x$ si y solo si $x: (Q1,S) \vdash^{*}_{M} (Q1,e)$.

Teorema 7.? Si L es un lenguaje sensible del contexto, entonces L es aceptado por un automatón lineal acotado.

Prueba: Si G = (Vn,Vt,F,S) es una gramática sensible del contexto construimos un autómata lineal acotado M tal que el lenguaje aceptado por M es L(G), la cinta de entrada contiene dos pistas. La pista 1 contiene la cadena de entrada x con el fin de marca, la pista 2 será usada para cálculos luego; M introduce una subrutina que ejecuta los siguientes pasos;

1.- La subrutina selecciona una cadena de simbolos consecutivos V de la pista 2, tal que V --- B es una producción en F.

2.- La subcadena V es reemplazada por B, recorriendo los símbolos que estan a la derecha de V si es necesario. Si esta operación causa que un símbolo sea impulsado hasta la derecha de fin de marca el autómata lineal acotado para.

3.- La subrutina elige indeterminísticamente ya sea regresar al paso uno o terminar.

En M"s salidas de la subrutina, la pista 1 contendrá alguna cadena U tal que

$$S \overset{*}{\underset{F}{===>}} U.$$

el autómata lineal acotado compara los símbolos de la pista 1 con los símbolos correspondientes de la pista 2. Si la comparación falla la cadena de símbolos sobre la pista 1 y 2 no son los mismos y el autómata lineal acotado para sin reconocer. Si las cadenas son las mismas el autómata lineal acotado para y acepta.

Teorema 2.6 Si A y B son convexos, también lo es su intersección.

Prueba: Sea $C = A \cap B$. Entonces

$f_C[tx1+(1-t)x2] = \min[f_A[tx1+(1-t)x2], f_B[tx1+(1-t)x2]]$

ahora puesto que A y B son convexos

$f_A(tx1+(1-t)x2] \geq \min[f_A(x1), f_A(x2)]$

$f_B(tx1+(1-t)x2] \geq \min[f_B(x1), f_B(x2)]$

de aqui

$$f_C[tx1+(1-t)x2] \geq \min[\min[f_A(x1), f_A(x2)], \min[f_B(x1), f_B(x2)]]$$

o equivalentemente

$$f_C[tx1+(1-t)x2] \geq \min[\min[f_A(x1), f_B(x1)], \min[f_A(x2), f_B(x2)]]$$

y así

$f_C[tx1+(1-t)x2] \geq \min[f_C(x1), f_C(x2)]$.

APENDICE B

FORMA NORMAL DE BACKUS

La forma normal de Backus FNB es una notación particular para las gramáticas libres del contexto, donde las reglas de producción de reescritura es representada por ::= en vez de --->, que ha sido utilizada en el capítulo 2. El conjunto de símbolos no terminales son representados por cadenas de caracteres encerrados en paréntesis triangulares "<", ">". Los terminales son directamente representados como se escriben. Una regla FNB para definir un no-terminal tiene el siguiente formato:

No-terminal,

seguido por el símbolo de producción "::=",

Seguido por una sucesión de alternativas,

separadas por el símbolo "|".

Cada alternativa es una sucesión de terminales (posiblemente vacía) y no-terminales.

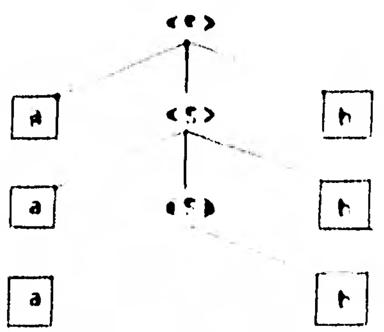
Un ejemplo para comparar las formas notacionales FNB y las gramáticas tipo 2 o libres del contexto de Chomsky para $\{ a^n b^n \mid n \geq 1 \}$ Es el siguiente:

FNB	gramáticas tipo 2 (Chomsky)
<S> ::= a<S>b	S ---> aSb
<S> ::= ab	S ---> ab

las alternativas usadas para combinar reglas donde "|" se puede leer como "o es un" permite que las reglas anteriores se puedan escribir como

<S> ::= a<S>b|ab

Finalmente, se dirá que el manejo de árboles de análisis gramatical en FNB es equivalente a los de las gramáticas de Chomsky, excepto que los no-terminales serán encerrados en paréntesis triangulares por ejemplo la FNB considerada anteriormente para n = 3 tendría el siguiente árbol de análisis gramatical:



GLOSARIO

Algoritmo: Es un procedimiento que siempre termina.

Constituyentes inmediatos: Los componentes de la lengua cuando esta se considera en su sustancia gráfica o fónica.

Contexto: Suma de la circunstancias, linguísticas o no que rodean a un hecho linguístico.

Elemento linguístico: Una o dos palabras que funcionan como una unidad y que no se pueden asignar a ninguna parte de la gramática.

Fonema: Clase de sonidos cuyas variantes solo estan condicionadas por el entorno fonético y no son significativas.

Fonología: Estudio del sistema en el que estan organizados los sonidos de la lengua. Incluye tanto la selección de ciertos sonidos entre las posibilidades existentes, como las relaciones entre ellos.

Léxico: El componente de la lengua(tanto gráfica como fonológicamente) no incluido en la gramática.

Morfema: la menor unidad significativa de la lengua, una palabra puede constituir en sí misma un morfema independiente; el morfema dependiente solo puede existir como parte de una palabra. En la palabra <<inmejorable>> hay dos morfemas dependientes <<in>> y <<able>> y uno independiente <<mejor>>. <<Morfema>> como <<fonema>>, hace referencia en ocasiones a una clase de unidades mínimas sin diferencia significativa por no ser intercambiables. Así, en <<imposible>>, <<no-tóxico>>,<<des-habitado>> y <<atónico>> <<im>>,<<no->>,<<des->>,<<a->> no se pueden intercambiar, por lo que se les debe considerar alomorfos de un morfema dependiente de negación.

Morfología: Estudio del sistema de variantes formales dentro de la lengua y su relación con otros aspectos de ella.

Oraciones-tipo básicas: Modelos oracionales a los que se pueden reducir todas la oraciones de una lengua.

Palabras gramaticales: Preposiciones,conjunciones, y verbos auxiliares que sirven para unir las palabras plenas en una secuencia con significado.

Palabras plenas: Nombres,verbos,adjetivos,etc.Que dotan de de contenido a la oración.

Procedimiento: Es una sucesión finita de instrucciones que pueden ser mecánicamente ejecutadas, tal como un programa de computadora.

Semántica: Estudio del significado desde el punto de vista mas conceptual que formal.

Sintagmática: la relación que se establece horizontalmente entre los elementos de una oración o entre las oraciones mismas.

Sintáxis: Estudio de la secuencia significativa de los elementos lingüísticos en el tiempo o en el espacio.

Latiz: Es un conjunto parcialmente ordenado tal que para cualquier pareja de elementos existe una mínima cota superior y una máxima cota inferior.

BIBLIOGRAFIA

A measurement-information discussion of fuzzy union and intersection
Ronald R. Yager
Int. J. Man-machine studies (1979) 11, 189,208

fuzzy sets
L. A. Zadeh
Information and control 8,338-353 (1965)

Some considerations on fuzzy automata
Masaharu Mizumoto,Jinichi Toyoda and Kohkichi Tanaka
electronics and communications in japan v.62-C, n.7, 1969

fuzzy automata and languages
Eugene Santos
information sciences 10, 193-197 (1976)

fuzzy languages
Masaharu Mizumoto,Jinichi Toyoda and Kohkichi Tanaka
system computer controls v.1, N.3, 1970

Quantitative fuzzy semantics
L .A. Zadeh
information sciences 3 (1971) 159-176

Deterministic acceptors of regular fuzzy languages
Michael G. Thomason and Peter N. Marinos
IEEE transactions on systems,man, and cybernetics march 1974

Pruf - a meaning representation language for natural language
L. A. Zadeh
int. J. Man-machine studies (1978) 10,395-460

Precisiation of human communication via traslation into pruf

L. A. Zadeh

A note on fuzzy sets
Joseph G. Brown
information and control 18, 32-39 (1971)

Finite fuzzy automata, regular fuzzy languaje, and pattern recognition
P. E. Thomason
pattern recognition 1973 v.5 383-390

On fuzzy mapping and control
Sheldon S.L. Lofti A. Zadeh
IEEE transaction on system man, and cybernetics smc-2 n.1,Jan,1972

Fuzzy algorithms
L.A.Zadeh
information and control 12,94-102(1968)

fuzzy-fuzzy automata
P. Mizumoto and K. Tanaka
kybernetes 1976,v.5,107-112

An formulation of fuzzy automata and its application as a model of learning systems
William G. Wee, K.S.Fu
IEEE transaction on system sciences and cybernetics, v. ssc-5 n.3,Jul 1979

Note on fuzzy languages
E.T. Lee and L.A. Zadeh
information sciences 1 (1969), 421-434

Learning of fuzzy formal languages
Shinichi Tamura and Kokichi Tanaka
IEEE transactions on system, man, and cybernetics january 1973

A general approach to linguistic approximation
F.Eshragh and E. H. Mamdani
int j. Man-machine studies (1979) 11,501-519

Linguistic models and fuzzy truths
Ronald R. Yager
int.J. Man-machine studies (1978) 10,483-494

Fuzzy languages and their relation to human and machine intelligence
L.A.Zadeh
man and computer proc.Int conf bordeaux 1970 130-165

On the correctness of semantic-sintax-directed traslation
Ramachandran Krishnaswamy and Artur B. Pyster
j. of the association for computing machinery v.27,N.2,April 1980,338-355

Concept representation in natural and artificial languages: Axioms,extension and aplications for fuzzy sets.
Joseph Goguen jr.
Int. J. Man-machine studies (1974) 6, 513-561

The design and analysis of computer algorithms
Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman

Currents in the theory of computing
Alfred V. Aho

La arboleda perdida
Rafael Alberti

Tree models for the description of languajes
Noam Chomsky

Theorie des ensembles flous vol 1
A. Kauffman

Aspects of language and language teaching
W. A. Bennett

Formal languages and their relations to automata
J.E. Hopcroft and J.D. Ullman
Reading, mass.: Addison-Wesley, 1969.

Grammars for programming languages
J.Craig Cleaveland, Robert C. Uzgalis
elsevier computer science library

introduccion en la linguistica general
John Lyons
Editorial teide/barcelona