EL USO DE REDES NEURALES PARA EL RECONOCIMIENTO DE FONEMAS

ING. GUALBERTO HIDALGO
ESCUELA POLITECNICA NACIONAL

ING. TANIA PEREZ
ESCUELA POLITECNICA NACIONAL

RESUMEN

En el presente artículo se explica en forma muy sucinta la utilización de redes neurales para el reconocimineto de fonemas. La red utilizada es un multiperceptrón de tres capas, de las cuales únicamente la segunda o intermedia y la de salida están constituidas por neuronas propiamente dichas. El algoritmo utilizado para el aprendizaje o entrenamiento de la red se conoce con el nombre de "retropropagación", es del tipo supervisado y consiste en el ajuste progresivo de los pesos de la red, proporcional al error entre el vector prefijado de antemano, y que recibe el nombre de vector objetivo, y el vector real calculado por la red. Al final se presentan algunos resultados.

ABSTRACT

In the present paper the use of neural networks for phoneme recognition is briefly explained. The used neural network is a three layer multiperceptron, with only the second and third layer formed by true neurons. The learning algorithm selected is "backpropagation", which is a supervised algorithm and consists in the progressive adjusment of the weights, proportional to the error between the prefixed (objective) and the actual vector given by the network. At the end some results are presented.

I. INTRODUCCION

Para captar las dificultades que conlleva el reconocimiento de fonemas resulta muy ilustrativo considerar las dos formas básicas a través de las cuales se realiza la intercomunicación humana: la palabra escrita y la palabra hablada. En lo que se refiere a la palabra escrita, si nos ceñimos a las características de la palabra impresa que predomina ampliamente sobre la manuscrita, podemos destacar en la misma las siguientes características: 1) un alto grado de normalización: los caracteres latinos de imprenta son de uso generalisado. Compárese la ventaja que esto representa frente al problema de entender la infinita variedad de escrituras individuales. 2) La discretización en virtud de la cual cada fonema aparece como un entidad independiente y claramente definida. Las separación entre palabras se destaca nitidamente a través de una separación espacial mayor. Como ayuda adicional están los signos de puntuación. La palabra hablada, en cambio, parece revestir caracteristicas diametralmente opuestas. Efectivamente, en tratándose de la palabra hablada, una normalización verdadera es un objetivo inalcanzable. Si bien un entrenamiento adecuado puede conferir un alto grado de inteligibilidad a ciertas voces, como es el caso de los locutores profesionales, nunca se podrá suprimir el conjunto de factores individuales que hacen de cada voz un fenómeno irrepetible. La infinita variedad de voces humanas constituye un universo tan rico que no existen dos voces que puedan decirse iguales. Aun si consideramos los fonemas emitidos por una misma persona, éstos están lejos de ser eventos estables. En efecto , incluso en este caso, la edad, el estado de ánimo, el estado de salud, etc. confieren a la pronunciación de cada fonema una matización siempre nueva e irrepetible.

Al contrario de lo que sucede con la palabra impresa en la que se ha llegado a un nivel claramente definido de discretización, la palabra hablada constituye un evento temporal continuo por excelencia. En el lenguaje hablado los fonemas dentro de cada palabra, e incluso entre palabras, se suceden unos a otros sin solución de continuidad interaccionando unos con otros, hasta el punto de que cada fonema reviste caracteres diferentes dependiendo de la interacción de los fonemas adyacentes. Este influjo mutuo de los fonemas adyacentes recibe el nombre de coarticulación. Es el que hace que una n entre dos vocales, por ejemplo, sea diferente de una n al final de palabra.

El alto grado de complejidad que conlleva el reconocimiento de la palabra hablada, ha dado origen a una serie de técnicas con las cuales se ha intentado resolver este difícil problema. Entre las mismas podemos mencionar la predicción lineal, el alabeo dinámico temporal, los modelos ocultos de Markov , las redes neurales. Hay que mencionar en todo caso que todas estas técnicas presuponen algún tipo de análisis espectral de la señal a reconocerse.

En el presente caso se intentó primeramente el reconocimiento de fonemas utilizando la predicción lineal, con resultados decididamente pobres. Ante este hecho se decidió utilizar la técnica de redes neurales con resultados que pueden considerarse prometedores.

En el presente artículo en la seccion II se exponen los pobres resultados obtenidos con la técnica de predicción lineal y la distancia de Itakura [1]. En la sección III se da una idea básica de las redes neurales. La sección IV presenta un análisis de la estructura de los multiperceptrones y se da una demostración detallada y estricta de la regla delta que que sirve para el entrenamiento "supervisado" de este tipo de redes. En la sección V se expone el preprocesamiento a que se somete la palabra hablada, y la forma de entrenamiento de redes

neurales para el reconocimiento de fonemas. La sección VI presenta algunos resultados y en la sección VII se extraen algunas conclusiones.

II. LA PREDICCION LINEAL Y LA DISTANCIA DE ITAKURA EN EL RECONOCIMIENTO DE FONEMAS

La técnica empleada originariamente para el reconocimiento de fonemas fue la predicción lineal [2]. Esta técnica asimila el aparato fonador a un filtro digital que al ser excitado apropiadamente genera los diversos fonemas del lenguaje hablado. Más concretamente, y en forma muy simplificada, el aparato fonador se representa por un sistema físico que puede ser excitado alternativamente por un tren periódico de impulsos o por ruido blanco aleatorio. Cuando la excitación es un tren de pulsos el fonema generado es de naturaleza periódica, por ejemplo una vocal. Si la excitación es ruido blanco aleatorio el sonido generado es una consonante sorda, como por ejemplo una s o una j. El modelo físico que representa el aparato fonador propiamente dicho se lo simula en un computador por medio de un filtro digital de tipo recursivo el cual tiene la forma indicada en la fig. 1.

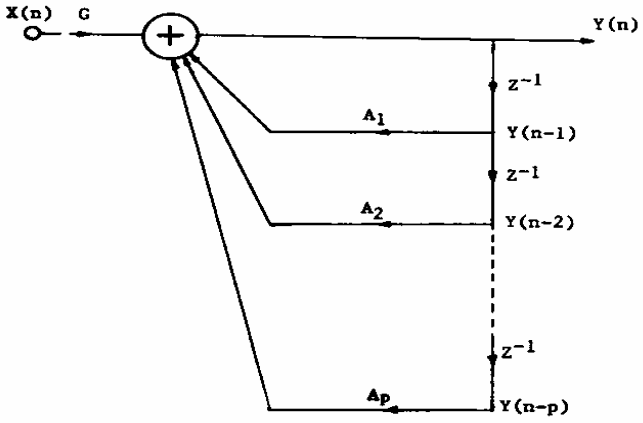

Fig. 1. Filtro digital recursivo.

Donde X(n) es la entrada o excitación

G es un factor de ganancia

Y(n) es la salida

y Z^{-1} representa un retardo temporal igual a un intervalo de muestreo T, que por simplicidad se considera unitario, esto es, T = 1 unidad de tiempo arbitraria.

La ecuación de diferencias que representa la salida del sistema está dada por la expresión:

$$Y(n) = GX(n) + A_1Y(n-1) + \ldots + A_pY(n-p)$$

donde p representa el orden del filtro.

Como se ve el valor de la salida en un instante nT = n (asumiendo T = 1) es función de la excitación X(n) y de las salidas de los p instantes previos. Esto justifica el nombre de recursivo para estos filtros, ya que la respuesta de los mismos se obtiene por una realimentación de la salida a la entrada. Los coeficientes que dosifican esta realimentación de las p salidas precedentes, reciben el nombre de coeficientes de predicción lineal y son los factores claves para el reconocimiento de fonemas. En efecto se puede considerar que los p coeficientes de predicción lineal: A_1, A_2, A_3, ..., A_p, que caracterizan un fonema, constituyen un vector en el espacio p-dimensional, en el cual cada vector, correspondiente a un fonema particular, tendrá una ubicación definida. Es evidente que aunque no existan 2 vectores que coincidan plenamente, esto es, dos pronunciaciones exactamente iguales de una vocal a, por ejemplo; los diversos vectores que correspondan a diferentes pronunciaciones de una misma vocal se ubicarán en una cierta vecindad en este espacio p-dimensional [3]. Tomando como base este hecho el reconocimiento de fonemas se lo realiza de la siguiente manera:

Primeramente se obtiene un muestrario de los diversos fonemas, por ejemplo de las vocales, y se calcula el "centroide" de cada una de las diversas clases de fonemas. En el caso de las vocales se tendrían 5 centroides correspondientes a las 5 vocales castellanas. El centroide de una vocal se caracteriza como aquel punto en el espacio p-dimensional que dista menos de todos los fonemas de su clase. Una vez determinados los centroides, la identificación de las vocales se hace por "distancia mínima", esto es, si se tiene un fonema desconocido, que podría ser cualquiera de las cinco vocales, se evaluará la distancia de dicha vocal a los centroides de las 5 vocales. La vocal desconocida se identificará con aquella a cuyo centroide diste menos. La "distancia" que se utiliza con este propósito es la distancia de Itakura, que recibe el nombre del investigador japonés que la propuso por primera vez. Esta distancia se expresa de diversas maneras, una de las más sencillas tiene la forma:

$$d(Ad,Ar) = (Ad - Ar)^T \phi (Ad - Ar)$$

donde Ad vector desconocido

Ar vector de referencia (centroide)

ϕ es la matriz de autocorrelación de los valores de las muestras del fonema cuyo reconocimiento se busca.

El vector Ar, que representa el centroide, se obtiene como el valor medio de los coeficientes de predicción lineal de cada una de las diversas clases de fonemas cuyo reconocimiento se busca. El cálculo del centroide para cada uno de los grupos de fonemas cuyo reconocimiento se pretende, constituye la etapa de entrenamiento del sistema. Para el caso de las cinco vocales se deberán calcular 5 centroides, 1 para cada vocal.

Aplicado este método al reconocimiento de vocales, los resultados obtenidos sin ser decepcionantes tampoco pudieron considerarse enteramente satisfactorios. En efecto mientras el reconocimiento de vocales como la a, la e, la i y la o tenía un altc porcentaje de acierto, el reconocimiento de la u resultó siempre pobre. En todo caso el porcentaje promedio global de acierto en el reconocimiento de las 5 vocales fluctuaba alrededor de un 90 %. Esto utilizando para el reconocimiento la misma población que sirvió para el entrenamiento.

Aunque los resultados no fueran tan promisorios como se hubiera deseado, se resolvió extender el método arriba descrito a los demás fonemas, agrupándolos en las siguientes categorías:

nasales: m, n
laterales: l, r
fricativas: f, j, s
oclusivas sonoras: b, d, g
oclusivas sordas: ch, k, p, t.

Los resultados obtenidos al aplicar la técnica descrita al reconocimiento de los grupos de fonemas arriba indicados fueron enteramente decepcionantes. El caso mas negativo fue el de las oclusivas sonoras: b, d, g, para las cuales se obtuvo un porcentaje de reconocimiento correcto inferior al 50%, y esto utilizando como población de prueba la misma utilizada para el entrenamiento.

Ante resultados tan pobres se decidió buscar alguna técnica más poderosa que garantizara un porcentaje más alto de aciertos.

Justamente por ese tiempo estaba cobrando gran actualidad la técnica denominada de "Redes Neurales", que se revelaba como especialmente apropiada para el reconocimiento de formas de gran riqueza de elementos y de características imprecisas y llenas de matizaciones como es el caso de la palabra hablada.

III. LAS REDES NEURALES: IDEAS BASICAS. [4]

Una red neural es una estructura computacional, cuyos modelos se han inspirado en en los procesos biológicos del sistema nervioso, considerando que, aún en sus formas más básicas un ser dotado de este sistema puede resolver problemas que ni siquiera las más avanzadas computadoras convencionales pueden hacerlo. La necesidad de modelar las funciones biológicas del sistema nervioso en una máquina realizada por el hombre ha obligado a profundizar en la comprensión de estos procesos biológicos.
El modelo básico de neurona que ha servido de inspiración para la creación de las redes neurales se muestra en la fig. 2 y consta de:
La célula nerviosa que es el cuerpo central de la neurona (soma).
El axón que está acoplado al soma y que es eléctricamente activo, ya que conduce los pulsos emitidos desde la neurona.
Las dendritas que son eléctricamente pasivas, pues reciben las pulsaciones enviadas desde otras neuronas por medio de un contacto especial denominado sinapsis.

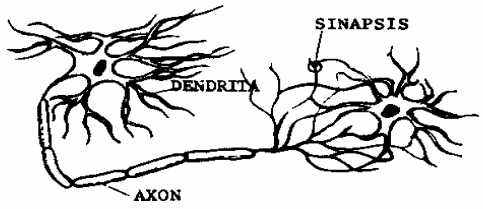

Fig. 2. Neurona Biológica

La sinapsis es la sutil conexión entre neuronas y es capaz de cambiar el potencial local de una dendrita en sentido positivo o negativo dependiendo del pulso que se está trasmitiendo.
En la fig. 3 se muestra un modelo simple de neurona "artificial" análoga al modelo biológico. En este caso las neuronas son elementos de procesamiento, mientras que los axones y dendritas podrían ser cables, y las sinapsis serían resistencias variables que llevan pesos a las entradas que representan datos o la suma de pesos de otros elementos de procesamiento.
Las entradas, que son voltajes proporcionales a los pesos establecidos se suman a través de los resistores. Cuando la suma alcanza un umbral predeterminado la neurona se activa, de lo contrario no lo hace.
Los elementos de procesamiento pueden interactuar de muchas maneras dependiendo de la forma como ellos están interconectados. La figura 3 muestra algunas posibilidades:

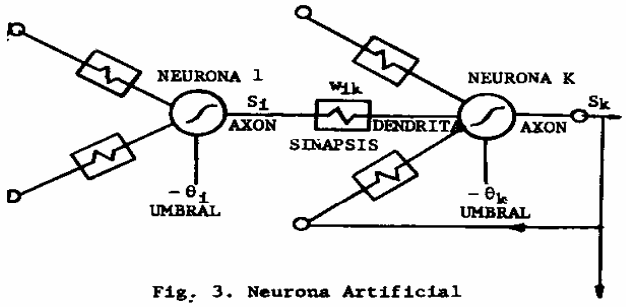

Fig. 3. Neurona Artificial

Elementos de procesamiento con o sin retropropagación.
Elementos de procesamiento conectados en forma completa a otros elementos de procesamiento y otros conectados parcialmente o enlazados solamente a alguno de ellos.

Entrenamiento

Las redes neurales, por definición son adaptivas o sujetas de entrenamiento. Existen muchas técnicas de entrenamiento (algoritmos), que pueden agruparse en tres categorías básicas:

Entrenamiento supervisado. Requiere la presencia de un "profesor" externo y del etiquetamiento de los datos usados para el entrenamiento de la red. El "profesor" conoce la respuesta correcta e ingresa una señal de error cuando la red produce una respuesta incorrecta. La señal de error "enseña" a la red la respuesta correcta y después de una sucesión de pruebas de aprendizaje, la red produce la respuesta deseada.

Entrenamiento no supervisado. Utiliza datos de entrenamiento no etiquetados y no requiere de un profesor externo. Los datos son entregados a la red, la cual forma grupos internamente y clasifica los datos en diferentes categorías.

Entrenamiento autosupervisado. Se utiliza con ciertas clases de redes neurales. El monitoreo de las redes se realiza internamente sin requerir de profesor externo. Una señal de error se genera en el sistema y se realimenta a la entrada del mismo. La respuesta correcta se produce después de un cierto número de interacciones.

Paralelismo

Si todos los elementos de una red están interconectados unos con otros, es decir cada elemento de procesamiento está enlazado a todos los demás elementos de procesamiento, se pueden efectuar miles de operaciones simples en forma paralela, encaminadas a la solución de un problema. Esta acumulación de muchas operaciones simples posibilita realizar operaciones complejas de clasificación, lo cual es esencial en las posibilidades que las redes neurales ofrecen.
La investigación de las redes neurales se ha profundizado últimamente debido fundamentalmente a tres factores: a) Un gran avance en las teorías matemáticas, b) El desarrollo de nuevas herramientas computacionales, y c) Los avances en la comprensión de los procesos neurobiológicos.

Limites en la simulación de redes neurales

El avance en el campo de estas redes ha desarrollado su propia terminología computacional la cual es importante comprender para poder valorar los requerimientos para la implementación y simulación de redes neuronales. Los conceptos y términos más utilizados son los siguientes:
Una red típica contiene muchas mas interconexiones que neuronas o elementos de procesamiento.
Cada interconexión requiere una o algunas operaciones acumulativas. Mientras que en una computadora digital la capacidad de almacenamiento o memoria se evalúa en términos de palabras (bytes) y la velocidad en términos de instrucciones por segundo; una red neural define al almacenamiento como el valor de los pesos de entrada y se mide en términos de interconexiones, y la velocidad en términos de interconexiones por segundo en una misma capa o entre diferentes capas.

Las herramientas computacionales actuales permiten una capacidad de simulación de aproximadamente 10^7 interconexiones por segundo, la cual es todavía lejana de la capacidad de la más modesta neurona biológica que alcanza velocidades del orden de 10^9 interconexiones por segundo.
En un futuro cercano se espera que nuevas tecnologías permitan incrementar las capacidades del hardware, de tal forma que los límites de almacenamiento y velocidad puedan extenderse significativamente. Algunas de las tecnologías más importantes son las siguientes:
El arseniuro de Galio (GaAs) en la fabricación de chips permitiría alcanzar la región de 10^{10} interconexiones por segundo. El mismo efecto se espera de la utilización de dispositivos de propósito especial con tecnología CCD (charge coupled device).
Para aumentar la capacidad de almacenamiento se continúan mejorando las tecnologías de la memoria RAM. Así mismo la tecnología de los chips tridimensionales se espera que podrían expandir las actuales capacidades de almacenamiento.
El multiprocesamiento de señales digitales puede contribuir a expandir tanto los límites de velocidad como los de almacenamiento.

Tecnologías de implementación de redes neurales

Las principales alternativas tecnológicas para la implementación directa de redes neurales incluyen las siguientes:
VLSI/VHSIC (very large scale integration/ very high speed integrated circuits). Esta tecnología está limitada a una baja densidad de interconexiones debido a su naturaleza bidimensional. Todos los pesos en una red implementada directamente con VLSI/VHSIC tendrían que almacenarse en memoria.

VLSI análogo. Tecnología ésta que está aún en desarrollo, y que, no obstante ser bidimensional, ofrece una alta densidad de interconexiones debido a que los pesos pueden ser implementados con resistencias obviando la necesidad de memoria adicional.

Tecnología óptica. Está todavía en desarrollo pero ofrece una altísima densidad de interconexiones debido a su naturaleza tridimensional.

Aplicaciones más relevantes y perspectivas de desarrollo de las redes neurales

Los diferentes modelos de redes neurales pueden realizar una gran variedad de funciones. Las más usuales están relacionadas con el reconocimiento de voz, procesamiento de imagen y problemas de robótica, entre otras.
Las tareas primarias más comunes incluyen clasificación de patrones, autoorganización o agrupamiento, almacenamiento de memoria asociativa y acceso a ella, preprocesamiento de visión y lenguaje , problemas de optimización computacional, procesamiento no lineal de señales, etc.

En el área de la imagen, los sistemas en tiempo real tienen una gran variedad de aplicaciones que incluyen las siguientes:

Segmentación de imagen;
Sistemas de seguridad basados en reconocimiento visual;
Detección automática y seguimiento de un objetivo;
Almacenamiento y trasmisión de imagen de alta calidad;
Ayudas para personas ciegas;
Análisis automático de diagnóstico médico en base de placas de rayos X;
Análisis automático de imágenes trasmitidas por satélite.

En el campo del reconocimiento de la voz, entre las tareas más importantes que pueden implementarse están las siguientes:

Transcripción automática;
Comunicación simplificada hombre- máquina;
Ayudas para personas sordas con salidas tangibles;
Ayudas para minusválidos que respondan a comandos de voz;
Reconocimiento de patrones de sonar;
Vigilancia acústica;
Traducción automática;
Almacenamiento y comunicación de voz de alta fidelidad;
Sistemas de seguridad con claves audibles;
Síntesis de texto a voz.

En el campo de la robótica actualmente se investiga para trasmitir a los robots la capacidad de aprendizaje y adaptabilidad de las redes neuronales. Los problemas más corrientes a ser enfrentados en este campo son:

Programación y control de las trayectorias de los brazos de los robots;
Coordinación brazo- cámara de video;
Fusión visión/tacto para reconocimiento de objetos.

En el área de Procesamiento de señales una aplicación muy importante en la que se investiga actualmente es en la recuperación de señales contaminadas de ruido o distorsionadas.

Entre las innumerables clases de redes neurales que han surgido y surgen cada día podemos mencionar las siguientes: las redes de Hopfield, los mapas autoorganizantes de Kohonen, los perceptrones y multiperceptrones, el neocognitrón, las memorias asociativas bidireccionales, etc.

De entre estas redes la que más ha sido utilizada para el reconocimiento de fonemas y de palabra hablada en general es el multiperceptrón.

IV. ESTRUCTURA DE UN MULTIPERCEPTRON [5]

Un multiperceptrón es una red neural estructurada de la siguiente manera:

1 capa de elementos de entrada, denominada capa de distribución por no estar constituida por neuronas propiamente dichas, sino sólo por elementos que distribuyen apropiadamente los valores o vectores de entrada.

1 o varias capas de neuronas intermedias que por su ubicación reciben el nombre de neuronas ocultas.

1 capa de neuronas de salida.

Las neuronas propiamente dichas, que forman las capas ocultas y la capa de salida, reciben también el nombre de elementos procesadores (EP), ya que realizan algún tipo de operación matemática sobre los valores que entran a las mismas. Específicamente cada neurona de una determinada capa suma todos los valores ponderados que le llegan de todas y cada una de las neuronas de la capa precedente y luego transmite a todas y cada una de las neuronas de la capa siguiente, el resultado de pasar esta suma a través de una función de activación F. La representación gráfica de una neurona o elemento procesador se da en la Fig. 4.

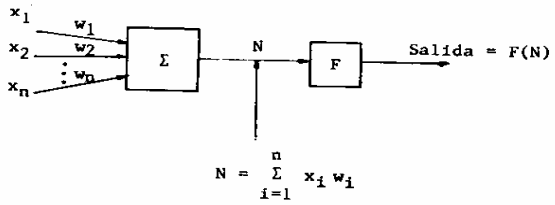

Figura 4. Neurona o elemento procesador (EP) en un multiperceptrón.

La función sigmoide [6]

La función de activación que más se utiliza en multiperceptrones es la función sigmoide cuya representación se da en la Fig. 5

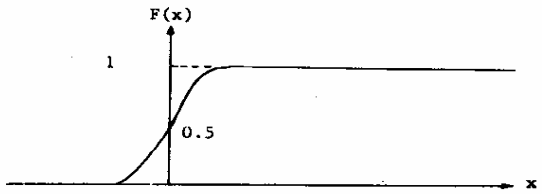

Figura 5. La función sigmoide.

La función sigmoide es una función monótonamente creciente que presenta cualidades magníficas para el entrenamiento o aprendizaje de las redes neurales. La definición matemática de la función sigmoide está dada por la expresión:

$$F(N) = 1/(1 + \exp(-N))$$

donde N es la suma ponderada de las salidas de la precedente capa de neuronas. Se dice suma ponderada porque cada salida de las neuronas de la capa precedente viene afectada de su correspondiente peso o factor de ponderación.

Como se desprende de la Fig. 5, la función sigmoide tiene las siguientes características:

a) Es una función derivable, lo cual nos permite obtener su derivada que se utilizará como un factor en el ajuste de los pesos, en el cual consiste el proceso de entrenamiento de las redes.

b) Sus características de ganancia son ideales para evitar problemas de saturación. En efecto, considerando la ganancia como el incremento de la función correspondiente a un incremento de la variable independiente, se ve que para esta función, pequeñas entradas, esto es valores de N en la vecindad de cero, producen la máxima ganancia, en tanto que entradas grandes, esto es valores de N que se aproximan a infinito generan ganancias que son prácticamente iguales a cero. De esta forma aun para valores de entrada que se aproximan a infinito negativo la función nunca desciende bajo cero, y para valores de entrada que tienden a infinito positivo la función nunca supera el valor de 1.

El umbral

De ordinario la salida de una neurona se calcula utilizando un umbral. Esto es, el argumento para la función sigmoide a la salida de la neurona toma la forma:

$$F(N) = 1/(1 + \exp-(N + \theta))$$

donde θ representa el valor del umbral. Esto produce un corrimiento en el origen de la función sigmoide y en esta forma acelera el proceso de convergencia durante el entrenamiento. Para incorporar este recurso en una red dada bastará con añadir a cada neurona un peso proveniente de una entrada con valor 1. Este peso se entrena de la misma manera que los restantes pesos, excepto que su entrada es siempre igual a 1 en vez de ser la salida de una neurona o elemento de distribución de la capa precedente.

La propagación de señales en un multiperceptrón se hace de entrada a salida sin que existan caminos de retroalimentación. Existen ramas unidireccionales de conexión, afectadas de sus correspondientes pesos, que van de cada una de las neuronas de una capa precedente a todas las de la siguiente.

Un multiperceptrón de 3 capas con tres elementos en la capa de distribución, dos neuronas intermedias u ocultas, y tres neuronas de salida se indica en la Fig. 6.

Funcionamiento de un multiperceptrón.

En el funcionamiento de un multiperceptrón se pueden distinguir dos actividades específicas: las de entrenamiento y las de reconocimiento. Durante el proceso de entrenamiento las salidas del multiperceptrón no corresponden a las deseadas, por lo tanto se obtiene un factor de corrección que es proporcional a este error, y se lo retropropaga para el ajuste de

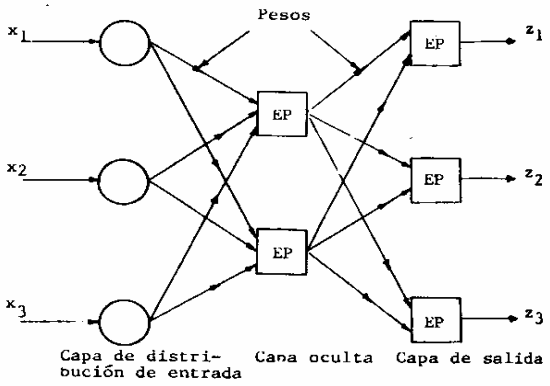

Figura 6. Multiperceptrón con una capa oculta.

los pesos. Este proceso se realiza en forma iterativa ya que el ajuste de los pesos se consigue en forma progresiva y no de una sola pasada.

Una vez que los pesos han sido ajustados, la red está preparada para el reconocimiento. Si se utiliza para este proceso la misma población que se utilizó para el entrenamiento, y si durante este último proceso se logró reducir el número de errores a cero, la red estará en capacidad de reconocer sin ningún error las formas presentadas. Si durante el proceso de entrenamiento no fue posible reducir el número de errores a cero, cabrá esperar un porcentaje de errores en el reconocimiento del mismo orden que el que se tuvo durante el entrenamiento.

Si la poblacion utilizada para el reconocimiento es diferente de la utilizada para el entrenamiento se deberá esperar un aumento del porcentaje de errores de reconocimiento respecto al porcentaje de errores obtenido durante el entrenamiento. Cuanto menor sea este aumento, tanto mejor será la capacidad de generalización de la red. En otras palabras, la generalización mide la capacidad que tiene una red de operar con formas no entrenadas, sin desmejorar el nivel alcanzado con las formas entrenadas. Para el proceso de reconocimiento se utilizará una sola de las etapas de que se compone una iteración en el proceso de entrenamiento, esto es, el "paso directo" del cual se habla más adelante.

Puesto que el entrenamiento de una red abarca como una de sus partes el conjunto de operaciones que se ejecutan para el reconocimiento de formas, bastará con hablar del entrenamiento de una red para tener una idea global de su forma de operar.

Entrenamiento de un multiperceptrón. [7]

El entrenamiento de un multiperceptrón pertenece al género de entrenamiento supervisado, esto es, aquel en que un "instructor" le indica a la red los resultados que debe arrojar, obligándole a rectificar procedimientos mientras tal cosa no ocurra.

El entrenamiento de una red consta de dos momentos claramente definidos:

1) El paso directo, que, como se ha dicho anteriormente, realizado una sola vez se aplica también al proceso de reconocimiento.
2) La retropropagación del error, que consiste en el ajuste paulatino de los pesos hasta que la red esté en capacidad de reconocer las formas que se le presentan.

EL paso directo.

El "paso directo" puede entenderse mejor si se toma como referencia la Fig. 7.

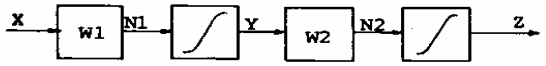

Figura 7. Diagrama de bloques para la explicación del paso directo.

En la Fig. 7 las letras X, N1, Y, N2, Z representan magnitudes vectoriales, mientras que W1 y W2 representan matrices. El vector de entrada X convenientemente distribuido por la capa de elementos de distribución de capa de entrada (que no aparece en la Fig. 7), entra en la capa oculta. En este momento se realiza el producto de la matriz W1 transpuesta, la matriz de los pesos que conectan la capa de distribución a la capa oculta, con el vector X. Este proceso genera un vector de salida N1 tal que

$$N1 = W1^T X$$

Los elementos de los vectores X y N1 y los de la matriz W1 se definen como sigue:

$X = \{x_h\}$ vector columna.

$h = 1, 2, \ldots\ldots\ldots$, Número de elementos en la capa de distribución. (1-NECD).

$W1 = \{w_{hi}\}$ Matriz.

$h = 1, 2, \ldots\ldots\ldots$, Número de elementos en la capa de distribución. (1-NECD).

$i = 1, 2, \ldots\ldots\ldots$, Número de neuronas en la capa oculta. (1-NNCO).

$N1 = \{n_i\}$ vector columna.

$i = 1, 2, \ldots\ldots\ldots$, Número de neuronas en la capa oculta. (1-NNCO).

El vector N1 al pasar por la función sigmoide se transforma en el vector Y, que constituye la salida de la capa oculta o intermedia, Y = F(N1), en donde Y representa un vector cuyos elementos se definen como,

$Y = \{y_i\}$ vector columna.
$i = 1, 2, \ldots\ldots\ldots$, Número de neuronas en la capa oculta (1-NNCO).

$$y_i = 1/(1 + \exp(-n_i))$$

(Para mayor simplicidad se asume que el umbral es siempre cero en las demostraciones que siguen, esta simplificación no altera los resultados).

Y es la entrada para la segunda capa de neuronas, que es la capa de salida. Aquí se realiza el producto de la matriz W2 transpuesta, la matriz de los pesos que van de la capa oculta a la capa de salida, con el vector Y, obteniéndose el vector producto N2, cuyo valor está dado por

$$N2 = W2^T Y$$

Los elementos de los vectores Y y N2 y los de la matriz W2 se definen como sigue:

$Y = \{y_i\}$ vector columna.

$i = 1, 2, \ldots\ldots\ldots$, Número de neuronas en la capa oculta. (1-NNCO).

$W2 = \{w_{ij}\}$ matriz.

$i = 1, 2, \ldots\ldots\ldots$, Número de neuronas en la capa oculta. (1-NNCO).

$j = 1, 2, \ldots\ldots\ldots$, Número de neuronas en la capa de salida. (1-NNCS).

$N2 = \{n_j\}$ $j = 1, 2, \ldots\ldots\ldots$, Número de neuronas en la capa de salida. (1-NNCS).

Al pasar el vector N2 por la función sigmoide se genera el vector Z, cuyos elementos se definen como sigue:

$Z = \{z_j\}$ vector columna.

$j = 1, 2, \ldots\ldots\ldots$, Número de neuronas en la capa de salida. (1-NNCS).

$$z_j = 1/(1 + \exp(-n_j))$$

Z es la salida de la red y debe ser igual al vector objetivo. Aquí se dan las siguientes posibilidades. Si el proceso en curso es un proceso de entrenamiento o de aprendizaje, se realiza la resta vectorial T - Z, en donde T es el vector objetivo y Z el vector de salida entregado por la red. Si el valor de esta resta vectorial arroja valores que superan los valores de un vector nivel o umbral prefijado, entonces comienza el proceso de

retropropagación de este error para el ajuste de los pesos. En caso contrario el proceso de entrenamiento termina. Si el proceso en curso es un proceso de reconocimiento, puesto que los pesos deben estar ya ajustados, o bien el vector de salida coincide con el vector objetivo, en cuyo caso ha tenido lugar un reconocimiento correcto, o bien el vector de salida no coincide con el vector objetivo, en cuyo caso se produce un error. Con esto se termina un "paso directo".

La Retropropagación.

Como se ha indicado más arriba este proceso sólo se realiza durante el entrenamiento y consiste en la propagación del error "hacia atrás" o sea de la salida a la entrada, de ahí el nombre de "retropropagación" con que se conoce este proceso. La regla que se usa con este propósito se denomina REGLA DELTA y consiste en realizar en forma iterativa ajustes o correciones en los pesos. Estos ajustes son proporcionales al error, esto es, a la diferencia entre el vector objetivo prefijado para la red y el valor del vector real de salida que ésta arroja.
Siguiendo un orden inverso, los pesos que se ajustan primero son los pesos que conectan la capa oculta con la capa de salida. Ya que en la capa de salida se dispone de un objetivo con un valor definido, el ajuste de los pesos se realiza fácilmente en este caso usando la regla delta que en esta oportunidad se obtiene de la siguiente manera:

La regla delta minimiza la suma de los cuadrados de las diferencias entre los valores de salida deseados y los calculados para todas las neuronas de salida y para todos los pares de vectores entrada-/salida. Esto es, se asume que el error para un patrón de entrada está dado por:

$$E = (1/2) \sum_{j=1}^{NNCS} (t_j - z_j)^2$$

donde j representa el orden de la neurona en capa de salida.

Para minimizar el error E se obtiene la derivada de E con respecto a los pesos w_{ij} que conectan la capa oculta a la capa de salida, donde el subíndice ij especifica el peso que va de la neurona i (en la capa oculta) a la neurona j en la capa de salida. Esto es, se obtiene el valor:

$$\frac{\partial E}{\partial w_{ij}}$$

Aplicando la regla de la cadena

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial z_j} \frac{\partial z_j}{\partial w_{ij}}$$

pero $z_j = \dfrac{1}{1 + e^{-n_j}}$

entonces

$$\frac{\partial z_j}{\partial w_{ij}} = \frac{\partial z_j}{\partial n_j} \frac{\partial n_j}{\partial w_{ij}}$$

enteramente explicitada $\frac{\partial E}{\partial w_{ij}}$ toma la forma

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial z_j} \frac{\partial z_j}{\partial n_j} \frac{\partial n_j}{\partial w_{ij}}$$

a partir de las definiciones

$$\frac{\partial E}{\partial z_j} = \frac{\partial}{\partial z_j} \left(\frac{1}{2} \sum_{j=1}^{NNCS} (t_j - z_j)^2 \right)$$

$$= - (t_j - z_j)$$

$$\frac{\partial z_j}{\partial n_j} = \frac{\partial}{\partial n_j} (1 + e^{-n_j})^{-1} = \frac{e^{-n_j}}{(1 + e^{-n_j})^2}$$

$$= \frac{1}{1 + e^{-n_j}} \quad \frac{e^{-n_j}}{1 + e^{-n_j}}$$

$$= z_j \left(1 - \frac{1}{1 + e^{-n_j}}\right)$$

$$= z_j (1 - z_j)$$

$$\frac{\partial n_j}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} \left(\sum_{i=1}^{NNCO} w_{ij} y_i \right) =$$

Juntando los 3 factores

$$\frac{\partial E}{\partial w_{ij}} = - z_j (1 - z_j)(t_j - z_j) y_i$$

donde:

z_j, salida de la neurona j en la capa de salida
t_j, objetivo para la neurona j en la capa de salida
y_i, salida de la neurona i en la capa oculta, conectada mediante el peso w_{ij} a la neurona j de la capa de salida.

Puesto que el tipo de corrección que se realiza es del tipo de descenso de gradiente, se debe tener

$$\Delta w_{ij} \propto \frac{-\partial E}{\partial w_{ij}}$$

Esto es, el ajuste a realizarse en el peso es proporcional al valor negativo de la derivada:

$$\frac{\partial E}{\partial w_{ij}}$$

Se introduce η como el factor de proporcionalidad conocido también como el factor de aprendizaje, que fija la velocidad de entrenamiento.

Consecuentemente el ajuste en el peso toma la forma

$$\Delta w_{ij} = \eta z_j (1 - z_j)(t_j - z_j) y_i$$

haciendo

$$(1 - z_j)(t_j - z_j) = \delta_j$$

$$\Delta w_{ij} = \eta\, \delta_j\, y_i$$

El valor del peso ajustado $w_{ij}(n+1)$ como función del peso antiguo (no ajustado) $w_{ij}(n)$ está dado por

$$w_{ij}(n+1) = w_{ij}(n) + \Delta w_{ij}$$

AJUSTE DE LOS PESOS EN LA CAPA OCULTA

La capa oculta no tiene vectores objetivo, por tanto en este caso no podrá usarse el ajuste descrito anteriormente. Fue precisamente la dificultad de encontrar un método apropiado de entrenamiento para las capas ocultas lo que obstaculizó el uso de los multipercentrones. La forma en que se entrenan las capas ocultas constituye el núcleo de la retropropagación.

Para la obtención del ajuste de los pesos en la capa oculta se usa el gráfico de la Fig. 8. En dicha figura:

$$n_i = \sum_{h=1}^{NECD} w_{hi}\, x_h \qquad i = 1, 2, \ldots ., NNCO$$

$$y_i = \frac{1}{1 + e^{-n_i}} \qquad i = 1, 2, \ldots ., NNCO$$

Aquí y_i representa la salida de cualquier neurona de la capa oculta. Se está tratando de ajustar cualquier peso que vaya del elemento h en la capa de distribución (h = 1, 2, ..., NECD) a la neurona i en la capa oculta.

Como se desprende de la Figura 8, un error en w_{hi} causa errores en cada una de las salidas de la red, z_j. El ajuste que se realice en w_{hi} debe tomar en cuenta todos estos errores. En otras palabras el ajuste Δw_{hi} debe ser proporcional al valor negativo de la suma de las derivadas:

$$\frac{\partial E_j}{\partial w_{hi}} \qquad j = 1, 2, \qquad NNCS$$

Esto es

$$\frac{\partial E}{\partial w_{hi}} = \frac{\partial E_1}{\partial w_{hi}} + \frac{\partial E_2}{\partial w_{hi}} + \ldots\ldots + \frac{\partial E_{NNCS}}{\partial w_{hi}}$$

$$= \sum_{j=1}^{NNCS} \frac{\partial E_j}{\partial w_{hi}}$$

donde

$$E_j = \frac{1}{2}(t_j - z_j)^2$$

Aplicando la regla de la cadena al término general:

$$\frac{\partial E_j}{\partial w_{hi}}$$

$$\frac{\partial E_j}{\partial w_{hi}} = \frac{\partial E_j}{\partial z_j}\frac{\partial z_j}{\partial n_j}\frac{\partial n_j}{\partial y_i}\frac{\partial y_i}{\partial n_i}\frac{\partial n_i}{\partial w_{hi}}$$

donde

$$\frac{\partial E_j}{\partial z_j} = -(t_j - z_j)$$

$$\frac{\partial z_j}{\partial n_j} = z_j(1 - z_j)$$

$$\frac{\partial n_j}{\partial y_i} = \frac{\partial}{\partial y_i}\left(\sum_{i=1}^{NNCO} w_{ij}\, y_i\right) = w_{ij}$$

$$\frac{\partial y_i}{\partial n_i} = \frac{\partial}{\partial n_i}\left(\frac{1}{1 + e^{-n_i}}\right);$$

$$= y_i(1 - y_i)$$

$$\frac{\partial n_i}{\partial w_{hi}} = \frac{\partial}{\partial w_{hi}}\left(\sum_{h=1}^{NECD} w_{hi}\, x_h\right) = x_h$$

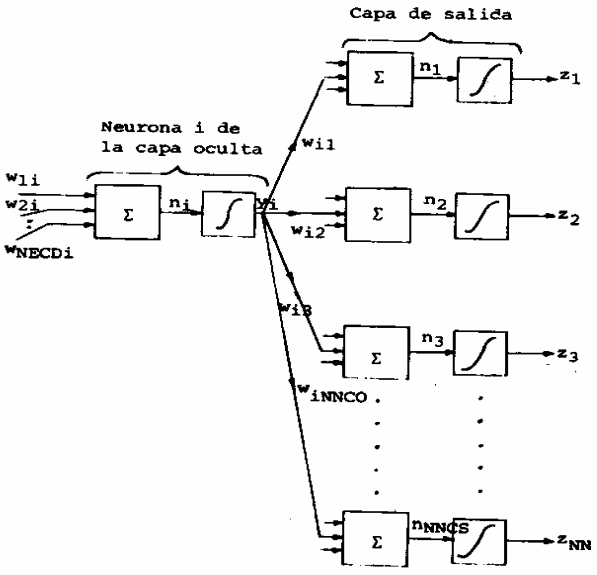

Figura 8. Gráfico para la obtención del ajuste de los pesos en la capa oculta.

Juntando los 5 factores

$$\frac{\partial E_j}{\partial w_{hi}} = -(t_j - z_j)\, z_j\, (1 - z_j)\, w_{ij}\, y_i\, (1 - y_i)\, x_h$$

Sumando de 1 a NNCS

$$\sum_{j=1}^{NNCS} \frac{\partial E_j}{\partial w_{hi}} =$$

$$= -y_i\, (1 - y_i)\, x_h \sum_{j=1}^{NNCS} (t_j - z_j)\, z_j\, (1 - z_j)\, w_{ij}$$

Anteriormente se ha hecho

$$(t_j - z_j)\, z_j\, (1 - z_j) = \delta_j$$

luego

$$\sum_{j=1}^{NNCS} \frac{\partial E_j}{\partial w_{hi}} = -y_i\, (1 - y_i) \left(\sum_{j=1}^{NNCS} \delta_j\, w_{ij} \right) x_h$$

haciendo

$$y_i\, (1 - y_i) \sum_{j=1}^{NNCS} \delta_j\, w_{ij} = \delta_i$$

$$\sum_{j=1}^{NNCS} \frac{\partial E_j}{\partial w_{hi}} = -\delta_i\, x_h$$

Teniendo en cuenta que el ajuste de w_{hi} debe ser proporcional al valor negativo de la suma de las derivadas que se acaban de obtener, Δw_{hi} tendrá la forma:

$$\Delta w_{hi} = \eta\, \delta_i\, x_h$$

donde

η, factor de proporcionalidad o factor de aprendizaje.

x_h, h-ésimo elemento del vector de entrada.

El valor del peso ajustado $w_{hi}(n+1)$ como función del peso antiguo (no ajustado) $w_{hi}(n)$ toma la forma:

$$w_{hi}(n+1) = w_{hi}(n) + \Delta w_{hi}$$

El proceso de entrenamiento consistirá entonces en introducir los datos cuyo reconocimiento se busca en forma secuencial en la red. Cada vez, y mientras la salida no coincida con el objetivo prefijado, se deberá retropropagar el error. Un paso directo seguido de la correspondiente retropropagación del error constituye una iteración en el proceso de aprendizaje.

V. PREPROCESAMIENTO DE LA PALABRA HABLADA Y ENTRENAMIENTO DE REDES NEURALES PARA EL RECONOCIMIENTO DE FONEMAS.

Para el entrenamiento de las redes en el reconocimiento de fonemas lo primero que debe conseguirse es una buena base de datos. Para esto se cuenta con un computador IBM PS/2 80 dotado de una tarjeta de adquisición de datos que permite muestrear y cuantizar el lenguaje hablado para luego pasarlo a un dispositivo de almacenamiento permanente que puede ser el disco duro del computador, o un disco flexible de 3.5" y de alta densidad. Si bien la tarjeta de adquisición de datos permite diferentes frecuencias de muestreo, para el presente trabajo sólo se utilizó la frecuencia de 12 KHz. Para la cuantización se utilizan 12 bits.

Las personas cuyas voces se utilizaron para el experimento son todas nativas ecuatorianas y todas del sexo masculino. Limitaciones de tiempo hicieron imposible la recolección de datos provenientes de voces femeninas. El número de personas que colaboraron en el experimento es de 58. El material fonético suministrado por estas personas varía grandemente. Así mientras una persona cuenta con un total de 38 archivos, existen 31 personas presentes con un solo archivo. Esto trae como consecuencia que el entrenamiento para las diversas voces sea muy desigual.

Puesto que exigir al computador una discriminación global de todos los fonemas del lenguaje ecuatoriano se consideró una meta demasiado exigente, se optó por agrupar los fonemas en conjuntos reducidos tomando en cuenta sus similitudes fonéticas. Como se dijo anteriormente los grupos escogidos fueron:

Vocales: a, e, i, o, u.

Oclusivas sonoras: b, d, g.

Oclusivas sordas: ch, k, p, t.

Fricativas: f, j, s.

Nasales: m, n.

Laterales: l, r.

Esta división lejos de considerarse definitiva se ha de juzgar como una primera tentativa de agrupamiento de fonemas para evaluar la capacidad de discriminación fonética de las redes neurales.

Se puede observar que esta clasificación es incompleta, en efecto faltan fonemas como la r inicial de palabra (o rr en medio de palabra), la ll, la ñ, los diptongos, etc.. En el grupo de las nasales habría que distinguir la n dentro de palabra y al final de ella.

Una vez en posesión de la base de datos antes mencionada se procedió a la segmentación de los fonemas cuyo entrenamiento interesaba. Esta segmentación se hizo de la siguiente manera:

Cada fonema segmentado tiene una longitud o duración de unos 250 milisegundos, que con una frecuencia de muestreo de 12 KHz

equivale a unas 3000 muestras. Se ha procurado ubicar el fonema analizado en la zona central de este segmento de 3000 muestras. La duración de los fonemas es en extremo variable. Así mientras hay algunos fonemas que copan casi por entero las 3000 muestras, otros apenas si son perceptibles por una ligera caída en el nivel energético en la zona de interés. Puesto que la duración del segmento de fonema procesado es la misma en todos los casos, se tendrá que los fonemas de corta duración se procesarán con su contexto, esto es, con la parte final del fonema que les precede y con la inicial del que les sigue, mientras que los fonemas de gran duración se procesarán solos, sin su contexto.

A fin de suministrar información temporal a las redes, el proceso de un fonema se hace de la siguiente manera:

De las 3000 muestras tomadas se procesan las 1300 centrales. Las 850 muestras iniciales y las 850 muestras finales se ignoran. La razón por la cual se tomaron 3000 muestras por segmento fue para tener libertad de variar a gusto la zona de procesamiento. De las 1300 muestras centrales se forman 4 bloques de 540 muestras. Cada bloque se corre respecto del anterior en 253 muestras. Este artificio podría compararse con el recurso que utiliza el cine de simular el movimiento a través de una secuencia de instantáneas en las que la posición de los objetos o personas en movimiento cambian ligeramente de una instantánea a otra. La Fig. 9 da una idea de la manera en que se forman los bloques para el procesamiento.

1300 muestras a procesarse

Muestra 850 a muestra 1390 Bloque 1
Muestra 1103 a muestra 1643 Bloque 2
Muestra 1356 a muestra 1896 Bloque 3
Muestra 1609 a muestra 2149 Bloque 4

Figura 9. Representación aproximada del corrimiento temporal de los cuatro bloques procesados de un fonema cualquiera.

Para ilustrar la forma en que se realiza el entrenamiento, supóngase que se dispone del mismo número N de vocales segmentadas. Entonces cada iteración consistirá en presentar al multiperceptrón las N pronunciaciones de las 5 vocales en la forma:

$(aeiou)_1, (aeiou)_2, \ldots\ldots, (aeiou)_N$

en donde cada subíndice corresponde a la n-ésima pronunciación de la vocal. Si se tiene un número diferente de vocales segmentadas, entonces el máximo subíndice N corresponderá al fonema con el menor número de pronunciaciones.

Con el fin de evaluar la capacidad de generalización de las redes, esto es, la posibilidad de las mismas de reconocer fonemas que no forman parte de la población de entrenamiento, se decidió utilizar los fonemas de numeración impar para el entrenamiento y los de numeración par como población de prueba. Por lo tanto el numero de fonemas utilizados para el entrenamiento de los diversos grupos es igual a [N/2], donde [N/2] significa el mayor entero contenido en N/2 y N el número de pronunciaciones del fonema menos pronunciado dentro de cada grupo.

De las experiencias acumuladas hasta el momento se ha llegado a la conclusión de que la forma en que se presentan los datos para el entrenamiento de redes puede crear diferencias abismales en la obtención de resultados.

En efecto en nuestro caso al comenzar las pruebas con redes neurales, primeramente se usaron como datos de entrada los coeficientes de reflexión (que son cantidades intermedias que sirven para la obtención de los coeficientes de predicción lineal), lo único que pudo comprobarse con esto fue la capacidad de las redes para aprender pero a un ritmo extremadamente lento. Posteriormente se utilizó como entrada para las redes el contenido energético de secciones del espectro de potencia de los fonemas, tal como son entregados por un banco de filtros ideales pasabanda escalonados en frecuencia de acuerdo a la escala de mel [8]. La Fig. 10 representa el espectro de potencia de la vocal a segmentado de acuerdo con la escala de mel. Para el presente caso se utilizan 17 filtros cuyo ancho de banda está dado por la separación entre dos verticales consecutivas de la figura.

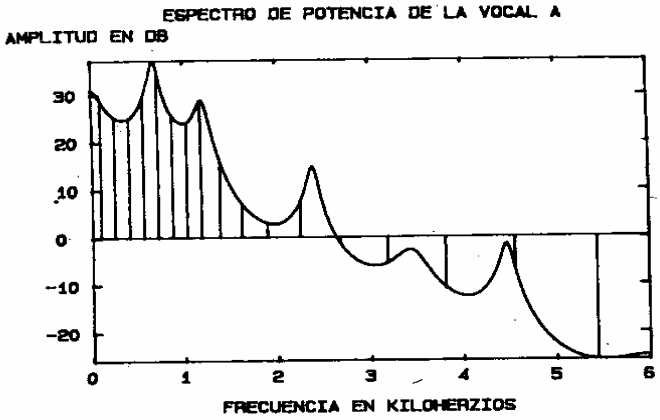

Figura 10. Espectro de potencia de la vocal a segmentado de acuerdo a la escala de mel.

Con este nuevo tipo de entrada los resultados mejoraron pero no en forma dramática.

Finalmente se utilizó como entrada para las redes, secciones escalonadas de acuerdo a la escala de mel, de la pendiente aproximada del espectro de potencia de los fonemas. La Fig. 11 presenta las salidas de los 17 filtros distribuidos de acuerdo a la escala de mel, cuando se utiliza la pendiente aproximada del espectro de potencia como magnitud de entrada para la excitación de las redes neurales.

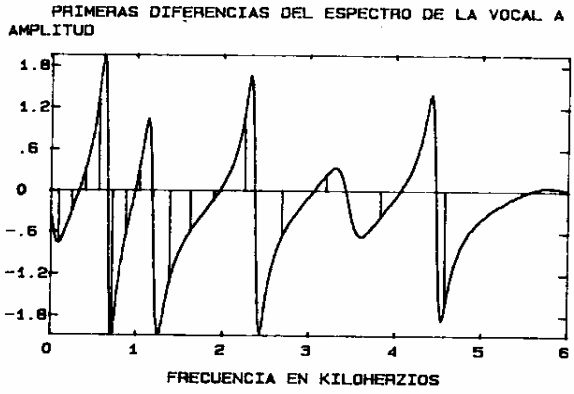

Figura 11. Gráfico de las pendientes aproximadas del espectro de potencia de la vocal a segmentada de acuerdo a la escala de mel.

Esta pendiente aproximada se la obtiene utilizando las primeras diferencias de dos valores consecutivos del espectro de potencia. En este último caso los resultados obtenidos fueron poco menos que espectaculares. Ver curvas de la Fig. 12

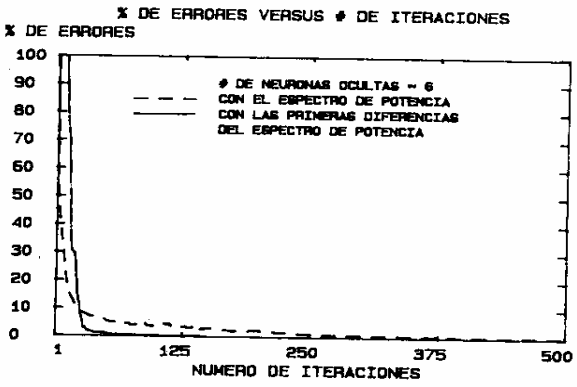

Figura 12. Diferencia en la velocidad de convergencia durante el aprendizaje cuando se utilizan como entrada para las redes neurales el espectro de potencia o las primeras diferencias del espectro de potencia.

En efecto como se desprende de dicha figura el método de la potencia muestra al comienzo un decrecimiento continuo de los errores, pero cuando llega a la vecindad de un 10% de errores, la convergencia se hace tan lenta que le toma aproximadamente 500 iteraciones para llegar al mismo procentaje de error (aproximadamente cero %) que con el método de las diferencias de potencia se alcanza en aproximadamente en 125 iteraciones. Cabe observar también que el método de las diferencias de potencia, al comienzo del entrenamiento, se mantiene en el máximo porcentaje posible de errores (100%) durante un buen número de iteraciones, y luego entra en un rápido proceso de convergencia de tal manera que necesita aproximadamente un cuarto del numero de iteraciones requeridas por el otro método para llegar a un nivel aproximado de cero % de errores.

Para obtener estos valores de entrada se procede de la siguiente manera:

1) Se obtienen 13 coeficientes de predicción lineal [2] de cada uno de los bloques del segmento a procesarse.

2) Se obtiene la transformada de Fourier de 512 puntos de los 13 coeficientes de predicción lineal.

3) A partir de la transformada de Fourier se obtiene el espectro de potencia de 256 puntos del fonema en cuestión.

4) Restando el término (I + 1) del término I se obtienen 255 puntos de la pendiente aproximada del espectro de potencia.

5) Estos 255 puntos de la pendiente aproximada del espectro de potencia se pasan a través de 17 filtros pasabanda ideales cuyas frecuencias de corte se distribuyen de acuerdo a la escala de mel [8].

El proceso que acabamos de describir se lo aplica a cada uno de los bloques de un fonema a procesarse. Puesto que por cada fonema se procesan 4 bloques y cada bloque entrega 17 magnitudes de salida, se tiene un conjunto de 68 elementos como vector de entrada.

Estructura de las redes para el procesamiento de cada uno de los grupos de fonemas.

Para facilitar la programación se buscó dar una estructura semejante a las redes utilizadas para el procesamiento de los diversos fonemas.

Como se mencionó anteriormente todas las redes utilizadas para el reconocimiento de fonemas tienen tres capas.

La primera capa recibe los 68 valores de entrada que resultan del procesamiento previo de los 4 bloques que se utilizan por fonema. Por lo tanto la primera capa estará constituida por 68 elementos de distribución.

Se ha demostrado que el número de neuronas para la capa intermedia u oculta debe ser igual al número de formas a reconocerse menos 1. Asumiendo que la redundancia podría garantizar una mayor confiabilidad en el funcionamiento de las redes, se prefirió para la capa oculta un número de neuronas igual al número de formas a reconocerse más 1. Así pues para la segunda capa se tiene: 6 neuronas para las vocales, 5 neuronas para las oclusivas sordas, 4 para las oclusivas sonoras, 4 para las fricativas, 3 para las nasales y 3 para las laterales.

Por fin en la tercera capa el número de neuronas es igual al número de formas a reconocerse. Por lo tanto se tienen 5 neuronas de salida para las vocales, 4 para las oclusivas sordas, 3 para las oclusivas sonoras, 3 para las fricativas, 2 para las nasales y 2 para las laterales.

El vector objetivo.

Como se ha indicado anteriormente el vector objetivo es aquel que debe arrojar la capa de salida de la red, y es aquel que se trata de conseguir a través del ajuste de los pesos. El vector objetivo utilizado en el presente trabajo tiene la forma que se indica a continuación.

Supóngase que se está trabajando con las cinco vocales y que se está tratando de obtener el reconocimiento de una de ellas. Supóngase además que las vocales están ordenadas siguiendo su secuencia normal dentro del alfabeto castellano, esto es, la secuencia aeiou. Por tanto las 5 neuronas de salida deberán ordenarse de tal manera que la primera neurona quede asignada a la a, la segunda a la e, y así sucesivamente hasta la quinta a la u.

Entonces cada vez que el fonema ingresado sea una a, el vector objetivo debe tener la forma: 1 0 0 0 0

Si el fonema ingresado es una e, el vector objetivo debe tener la forma: 0 1 0 0 0.

Y así sucesivamente hasta la vocal u, cuyo vector objetivo debe tener la forma: 0 0 0 0 1.

Teniendo en cuenta que el reconocimiento se hace sólo por mayoría, esto es, identificando el fonema de ingreso con la neurona que arroja el mayor valor de salida, se aproximan los ceros por cualquier cantidad igual o menor que 0.1 y los unos por cualquier cantidad igual o mayor que 0.9. Este recurso acorta notablemente el tiempo de entrenamiento, ya que llegar a los valores cero y uno consumiría un tiempo excesivo de computación.
Todo aquel que ha entrado en contacto con la técnica de redes neurales queda sorprendido de la inagotable variedad de posibilidades que éstas ofrecen. Uno de los hechos más de lamentar es la imposibilidad en que uno se encuentra de probarlas todas. Dentro de la técnica de retropropagación utilizada en el presente trabajo existirían, por ejemplo, otras formas de codificar el vector objetivo de salida. Se podría aumentar el numero de neuronas ocultas. Se podría estructurar el vector de entrada con un mayor o un menor número de bloques a procesarse, lo cual conduciría a una variación de los elementos de distribución en la capa de entrada. Se ha renunciado a estas posibilidades por limitaciones de tiempo, sin embargo si se implementaron 4 variaciones de la técnica de retropropagación que las denominaremos:

1) Técnica sin umbrales, que equipara a cero el valor de todos los umbrales.

2) Técnica con umbrales, que, a diferencia de la técnica anterior, no prescinde de los umbrales.

3) Técnica con umbrales y momentos, que, además de los umbrales, introduce en el ajuste presente de los pesos una cierta magnitud ponderada del ajuste precedente.

4) Técnica de Sejnowsky Y Rosenberg que puede considerarse una variación de la técnica precedente, estando constituido el ajuste del peso por el ajuste normal afectado por el factor $(1 - \alpha)$ más el ajuste inmediatamente precedente afectado por el factor α. La variación de α va de 0 a 1.

VI. RESULTADOS

A continuación se presentan los porcentajes promedio de acierto de estas 4 técnicas aplicadas a los 6 grupos de fonemas establecidos para el experimento:

Cuadro comparativo de los 4 métodos para el reconocimiento de las vocales. La abreviación "P." significa población.

	P. entrenada % aciertos	P. no entrenada % aciertos
Sin umbrales	99.53	97.18
Con umbrales	99.53	95.29
Con umbrales y momentos	99.53	95.06
Sejnowsky y Rosenberg	99.76	95.53

Cuadro comparativo de los 4 métodos para el reconocimiento de las consonantes oclusivas sordas: ch, k, p, t.

	P. entrenada % aciertos	P. no entrenada % aciertos
Sin umbrales	99.56	86.40
Con umbrales	99.56	92.11
Con umbrales y momentos	99.56	92.11
Sejnowsky y Rosenberg	99.12	92.54

Cuadro comparativo de los 4 métodos para el reconocimiento de las consonantes oclusivas sonoras: b, d, g.

	P. entrenada % aciertos	P. no entrenada % aciertos
Sin umbrales	97.14	73.81
Con umbrales	97.14	74.76
Con umbrales y momentos	97.14	75.24
Sejnowsky y Rosenberg	97.14	74.29

Cuadro comparativo de los 4 métodos para el reconocimiento de las consonantes fricativas: f, j, s.

	P. entrenada % aciertos	P. no entrenada % aciertos
Sin umbrales	95.16	78.49
Con umbrales	98.92	70.43
Con umbrales y momentos	98.39	70.97
Sejnowsky y Rosenberg	98.92	75.27

Cuadro comparativo de los 4 métodos para el reconocimiento de las consonantes nasales: m, n.

	P. entrenada % aciertos	P. no entrenada % aciertos
Sin umbrales	98.67	78
Con umbrales	98.67	75.33
Con umbrales y momentos	98.67	78.67
Sejnowsky y Rosenberg	98	75.33

Cuadro comparativo de los 4 métodos para el reconocimiento de las consonantes laterales: l, r.

	P. entrenada % aciertos	P. no entrenada % aciertos
Sin umbrales	99.44	76.67
Con umbrales	98.33	77.78
Con umbrales y momentos	*96.67*	*78.33*
Sejnowsky y Rosenberg	97.22	77.78

VII. CONCLUSIONES Y PERSPECTIVAS.

De las tablas de valores presentadas en la sección anterior se puede extraer el siguiente cuadro de valores promedio y desviaciones estándar:

VOCALES: Población entrenada

Valor promedio	Desviación estándar
99.5875	0.1150017

VOCALES: Población no entrenada

Valor promedio	Desviación estándar
95.765	0.9626533

OCLUSIVAS SORDAS: Población entrenada

Valor promedio	Desviación estándar
99.45	0.2199974

OCLUSIVAS SORDAS: Población no entrenada

Valor promedio	Desviación estándar
90.79	2.933678

OCLUSIVAS SONORAS: Población entrenada

Valor promedio	Desviación estándar
97.14	0

OCLUSIVAS SONORAS: Población no entrenad

Valor promedio	Desviación estándar
74.525	0.6145191

FRICATIVAS: Población entrenada

Valor promedio	Desviación estándar
97.8475	1.809

FRICATIVAS: Población no entrenada

Valor promedio	Desviación estándar
73.79	3.808865

NASALES: Población entrenada

Valor promedio	Desviación estándar
98.5025	0.3349991

NASALES: Poblacion no entrenada

Valor promedio	Desviación estándar
76.8325	1.756365

LATERALES: Población entrenada

Valor promedio	Desviación estándar
97.915	1.228944

LATERALES: Población no entrenada

Valor promedio	*Desviación estánda*
77.64	*0.6967078*

De los valores presentados se pueden extraer las siguientes conclusiones:

1) El reconocimiento de vocales es notablemente alto. Para la población entrenada bordea el 100%. El reconocimiento la población no entrenada no muestra deterioro notable respecto de la entrenada, lo cual demuestra una excelente capacidad de generalización de las redes.

2) El reconocimiento de las oclusiv sordas muestra un grado de acierto aceptable y el deterioro en el reconocimiento de la población no entrenada respecto de la entrenada no es muy signi cativo.

3) En los restantes 4 grupos el reconocimiento de la población entrenada muest un alto grado de confiabilidad, sin bargo en la población no entrenada observa un deterioro notorio, que en 4 casos se traduce en una caída del porcentaje de aciertos de un 20% respecto la población entrenada. Una posible plicación de este deterioro en la capacidad de generalización de la red hab que encontrarla en el hecho de que, si do estos fonemas altamente dependien del contexto, requieren de un entrenamiento mucho mas rico, tal que abarque mayor parte de los contextos en los estos fonemas pueden ocurrir.
Para superar estas limitaciones se imp la búsqueda de nuevos caminos, que v

tajosamente pueden encontrarse dentro de las innumerables posibilidades que las redes neurales ofrecen. En efecto, éstas presentan nuevas perspectivas en el área del reconocimiento de voz. Al menos dos se vislumbran como las ventajas potenciales más importantes sobre los métodos ya existentes.

Primera: las altas velocidades que pueden alcanzar en los procesos computacionales, son las requeridas para el reconocimiento de palabra hablada continua, mediante la utilización de muchos elementos simples operando en paralelo.

Segunda: los nuevos algoritmos con redes neurales son capaces de adaptar los parámetros internos de las redes al mismo tiempo que se autoorganizan y optimizan la ejecución. Especialmente importantes resultan estas posibilidades para aplicaciones tales como reconocimiento de palabra hablada dependiente e independiente del locutor.

Algunos investigadores han propuesto nuevas formas de preprocesamiento inspiradas en la fisiología del oído humano y en los resultados de la psicoacústica. Al menos tres son los modelos que están estudiándose bajo esta nueva perspectiva, y que para el reconocimiento toman en cuenta factores tales como: la mayor o menor frecuencia de las palabras, la posibilidad de palabras inapropiadas contextualmente, la adición de información de coarticulación en el reconocimiento, así como la posibilidad de tartamudeo y de cambios en el ritmo de la voz. Todos estos factores, que no se consideran en los modelos actuales de reconocimiento, son esenciales en la comunicación humana. Por lo tanto, mientras ellos sean ignorados por las máquinas, el éxito en el cumplimiento de esta tarea será relativo.

La enorme complejidad que representa el reconocimiento de voz , en todas sus formas, con un porcentaje de errores aceptable, mantendrá actual la investigación en este campo aun por muchos años más.

REFERENCIAS.

[1] F. Itakura, "Minimum Prediction Residual Principle Applied to Speech Recognition," IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-23, No. 1, February 1975, pp. 67-72.

[2] J. Makhoul, "Linear Predition: A Tutorial Review," IEEE Proceedings, Vol. 63, No. 4, pp. 550-580, April 1975.

[3] J. Makhoul, S. Roucos, and G. Gish, "Vector Quantisation in Speech Coding," IEEE Proceedings, Vol. 73, No. 11, November 1985, pp. 1551-1588.

[4] DARPA. Neural Network Study, Published by AFCEA International

[5] Philip D. Wasserman, Neural Computing Theory and Practice, Van Nostrand Reinhold, New York, 1989.

[6] P. K. Simpson, "Foundations of neural networks," En Artificial Neural Networks, E. Sánchez Sinencio and C. Lau, Eds., IEEE Press, 1992.

[7] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Parallel Distributed Processing, vol. 1, Cambridge, MA: M.I.T. Press,1986.

[8] A. Waibel and B. Yegnanarayana, "Comparative study of nonlinear time warping techniques in isolated word speech recognition systems," Tech. Rep. Carnegie Mellon University. June 1981.

HIDALGO, GUALBERTO

Ingeniero en Electrónica y Telecomunicaciones graduado en la Escuela Politécnica Nacional en el año de 1974. Obtuvo el título de Master en Ingeniería de Comunicaciones en el Imperial College de Londres, Inglaterra, en 1979. Actualmente se desempeña como profesor a tiempo completo en la Escuela Politécnica Nacional y es estudiante externo de la Universidad de Londres. Dirige el PROYECTO CONUEP 88-01, RECONOCIMIENTO DE FONEMAS POR COMPUTADOR.

PEREZ, TANIA

Ingeniero en Electrónica y Telecomunicaciones graduada en el Instituto Bonch Bruyevich, Leningrado, 1977. Actualmente se desempeña como profesora principal a tiempo completo en la Escuela Politécnica Nacional y realiza estudios de postgrado en Computación e Informática en la misma Institución. Participa en el PROYECTO CONUEP 88-01, RECONOCIMIENTO DE FONEMAS POR COMPUTADOR.