Reconocimiento de Voz usando Redes Neuronales Artificiales Backpropagation y Coeficientes LPC

Luis. A. Cruz-Beltrán[1] and Marco. A. Acevedo-Mosqueda[1]

[1] SEPI-Telecomunicaciones ESIME IPN Unidad Profesional "Adolfo López Mateos". Col. Lindavista, 07738, México. D. F
lcruzb06@ipn.mx, macevedo@ipn.mx

Abstract. En esta investigación se propone un algoritmo para el reconocimiento de personas en un canal telefónico. El algoritmo se basa en el comportamiento de las Redes Neuronales Artificiales (RNA), en particular, sobre el algoritmo Backpropagation. En este trabajo se utilizan archivos de voz con formato estándar *.Wav. Se propone un método para la identificación de hablantes en el canal telefónico. Presentamos los pasos a seguir para la identificación de hablantes en el canal telefónico que nos brindaron los mejores resultados en varias pruebas que realizamos sobre el método que propusimos para esta investigación.

Keywords: RNA, Wavelets, Backpropagation, Código de Predicción Lineal (LPC), Wav.

1 Introducción

La verificación o identificación de personas en un canal telefónico, empleando su patrón de voz, es la base para la realización de este artículo en el cual se propone un sistema que emplea las características del patrón de voz, LPC y RNA. Para poder solucionar eficientemente un caso jurídico en el cual se encuentra inmiscuida una grabación telefónica del acusado como prueba del caso y se necesita de un sistema que autentifique y corrobore si efectivamente la voz que se encuentra en la grabación pertenece al inculpado que se encuentra en el proceso penal.

Este documento se divide en las siguientes secciones. En la Sección 2 se presenta un panorama general de las RNA y los coeficientes LPC. La Sección 3 muestra la arquitectura de la RNA Backpropagation. En la Sección 4 se da una explicación detallada del sistema propuesto. En la Sección 5 se presentan los resultados obtenidos. Finalmente, la Sección 6 es dedicada a las conclusiones de este artículo.

M.A Cruz-Chávez, J.C. Zavala-Díaz (Eds): CICos 2008, ISBN:978-607-00-0165-9. pp. 89-99, 2008

2 Redes Neuronales Artificiales

Las RNA son sistemas de procesamiento de información cuya estructura y funcionamiento están inspirados en las redes neuronales biológicas [1]. En todo modelo de RNA se tienen cuatro elementos básicos.

- Un conjunto de conexiones, pesos ó sinapsis que determinan el comportamiento de la neurona, las cuales pueden ser excitadoras, presentan un signo positivo (conexiones positivas) y las inhibidoras presentan un signo negativo (conexiones negativas).
- Una función que se encarga de sumar todas las entradas multiplicadas por sus pesos correspondientes.
- Una función de activación que puede ser lineal ó no lineal empleada para limitar la amplitud de la salida de la neurona.
- Una ganancia exterior que determina el umbral de activación de la neurona.

Desde que el psicólogo Frank Rosenblatt en 1957 [1] introdujo el modelo del perceptrón de una sola capa, las RNA se convirtieron en una herramienta poderosa para solucionar diversos tipos de problemas relacionados con la clasificación, estimación funcional y optimización del reconocimiento de patrones.

El modelo propuesto se observa en (1), donde $x_{p1},\ldots, x_{pi}$ son las unidades de entrada, $w_{j1},\ldots,w_{ji}$ son los pesos de la RNA, b_i es la ganancia ó umbral de activación, N_{pj} es el producto de los pesos con respecto a la entrada, f es la función de activación de la RNA y finalmente y_{pj} es la salida de la RNA, estas variables se relacionan en la siguiente expresión:

$$y_{pj} = f(N_{pj} = \sum_{i=1}^{m} x_{pi} w_{ji} + b_i), \quad para \quad m \in IR,\ m < \infty. \tag{1}$$

2.1 Código de Predicción Lineal

Una gran parte de las aplicaciones relacionadas con el tratamiento del habla, están basadas en el análisis de LPC, dado que es capaz de extraer la información lingüística y eliminar la correspondiente a la persona en particular. La predicción lineal modela la zona vocal humana como una respuesta al impulso infinita, que produzca la señal de voz.

El término predicción lineal se refiere al método para predecir ó aproximar una muestra de una señal en el dominio del tiempo *s[n]* basada en varias muestras anteriores *s[n - 1], s[n - 2], s[n -M].*

$$s[n] \approx \hat{s}[n] = -\sum_{i=1}^{M} a_i s[n-i]. \tag{2}$$

donde *s[n]* es llamada señal muestreada, y a_i, i = 1,2,…,M son los predictores ó coeficientes LPC. Un pequeño número de coeficientes LPC $a_1, a_2,\ldots,a_M$ pueden ser

usados para representar eficientemente una señal *s[n]*[3,6]. Los valores $a_1, a_2,\ldots,a_M$ son la base para la realización de este trabajo debido a que nos ayudan a modelar los parámetros de la voz de cada uno de los hablantes que se emplean en este sistema propuesto.

3 La Red Backpropagation

En 1986, Rumelhart, Hinton y Williams, basados en otros trabajos formalizaron un método para que una red neuronal aprendiera la asociación que existe entre los patrones de entrada a la misma y las clases correspondientes, utilizando más niveles de neuronas que los que utilizó Rosenblatt para desarrollar el Perceptrón.

Este nuevo método es conocido como Backpropagation (retropropagación del error) que es un tipo de red con aprendizaje supervisado, el cual emplea un ciclo propagación-adaptación de dos fases.

Una vez aplicado un patrón de entrenamiento a la entrada de la red, este se propaga desde la primera capa a través de las capas subsecuentes de la red, hasta generar una salida, la cual es comparada con la salida deseada y se calcula una señal de error para cada una de las salidas, a su vez esta es propagada hacia atrás, empezando de la capa de salida, hacia todas las capas de la red hasta llegar a la capa de entrada, con la finalidad de actualizar los pesos de conexión de cada neurona, para hacer que la red converja a un estado que le permita clasificar correctamente todos los patrones de entrenamiento. La estructura general se muestra en la Figura. 1.

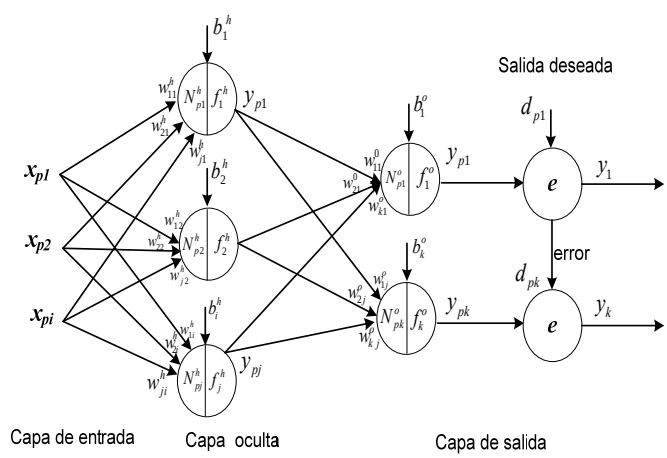

Fig. 1. Modelo de la RNA Backpropagation.

3.1 Algoritmo de entrenamiento de la Red.

A continuación se presenta el algoritmo empleado para el entrenamiento de la RNA Backpropagation. [1][2][5].

1. Inicializar los pesos de la red (w) con valores aleatorios pequeños.
2. Mientras la condición de paro sea falsa realizar los pasos (3-6).
3. Se presenta un patrón de entrada, ($x_{p1}, x_{p2},\ldots,x_{pi}$) y se específica la salida deseada que debe generar la red ($d_{p1}, d_{p2},\ldots,d_{pk}$).
4. Se calcula la salida actual de la red, para ello se presentan las entradas a la red y se va calculando la salida que presenta cada capa hasta llegar a la capa de salida ($y_1,y_2,\ldots,y_k$). Los pasos son los siguientes:
   a) Se determinan las entradas netas para las neuronas ocultas procedentes de las neuronas de entrada.

$$N_{pj}^{h} = \sum_{i=1}^{m} w_{ji}^{h} x_{pi} + b_{i}^{h}. \tag{3}$$

   b) Se aplica la función de activación a cada una de las entradas de la neurona oculta para obtener su respectiva salida.

$$y_{pj} = f_{j}^{h}(N_{pj}^{h} = \sum_{i=1}^{m} w_{ji}^{h} x_{pi} + b_{i}^{h}). \tag{4}$$

   c) Se realizan los mismos cálculos para obtener las respectivas salidas de las neuronas de la capa de salida.

$$\begin{aligned} N_{pk}^{o} &= \sum_{j=1}^{m} w_{kj}^{o} y_{pj} + b_{k}^{o}; \\ y_{pk} &= f_{k}^{o}(N_{pk}^{o} = \sum_{j=1}^{m} w_{kj}^{o} y_{pj} + b_{k}^{o}). \end{aligned} \tag{5}$$

5. Determinación de los términos de error para todas las neuronas:
   a) Cálculo del error (salida deseada–salida obtenida).

$$e = (d_{pk} - y_{pk}). \tag{6}$$

   b) Obtención de la delta (producto del error con la derivada de la función de activación con respecto a los pesos de la red).

$$\delta_{pk}^{o} = e * f_{k}^{o'}(N_{pk}^{o}). \tag{7}$$

6. Actualización de los pesos. Se emplea el algoritmo recursivo del gradiente descendente, comenzando por las neuronas de salida y trabajando hacia atrás hasta llegar a la capa de entrada.

a) Para los pesos de las neuronas de la capa de salida:

$$w_{kj}^{o}(t+1) = w_{kj}^{o}(t) + \Delta w_{kj}^{o}(t+1);$$
$$\Delta w_{kj}^{o}(t+1) = miu\delta_{pk}^{o} y_{pj}. \quad (8)$$

b) Para los pesos de las neuronas de la capa oculta:

$$w_{ji}^{h}(t+1) = w_{ji}^{h}(t) + \Delta w_{ji}^{h}(t+1);$$
$$\Delta w_{ji}^{h}(t+1) = miu\delta_{pj}^{h} x_{pi}. \quad (9)$$

7. Se cumple la condición de paro (error mínimo ó número de iteraciones alcanzado logrado).

4 Algoritmo Empleado

La Figura. 2, muestra el sistema propuesto, el cual consiste de tres etapas: la etapa de la captura de la señal de voz del canal telefónico, la etapa de preprocesamiento de la señal y finalmente la etapa de verificación del hablante empleando las características extraídas en las dos primeras etapas.

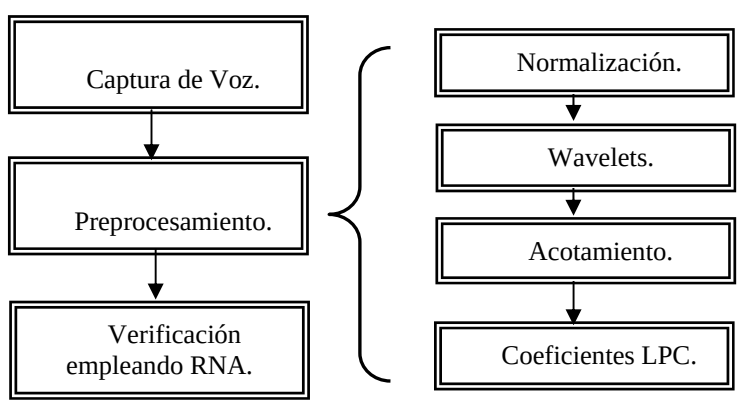

Fig. 2. Sistema Propuesto.

4.1 Captura de Voz

Para la realización de la captura de la voz se propone registrar 5 veces la frase "Zoológico" con cinco personas diferentes, cada una de ellas grabó la misma frase con estados de ánimo diferentes. Los hablantes fueron Luis, Orlando, Alejandro, Diana y Leydi de 23, 29, 30, 5 y 22 años de edad, respectivamente.

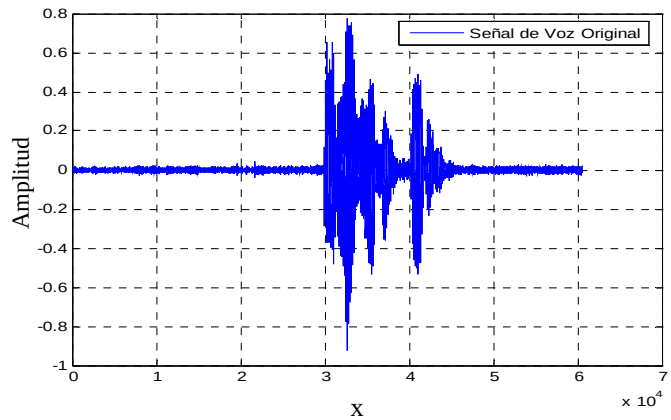

Fig. 3. Señal de voz grabada**.**

Se escogió la palabra “Zoológico” debido a que esta contiene la mayor parte de los formantes de la voz, gran cantidad de características espectrales. El procedimiento se llevó a cabo de la siguiente manera: Se instaló el software Mercury en una PC conectada a la línea telefónica vía MODEM. Después, cada uno de los 5 hablantes realizó una llamada telefónica al número de la casa donde se encontraba conectada la PC a la línea telefónica. Con el software Mercury se grabó en la PC la conversación con la palabra Zoológico pronunciada desde la caseta telefónica. La conversación contiene el ruido ambiental y el ruido del canal telefónico. Este proceso se realizó 5 veces para cada una de los 5 hablantes, obteniendo así 25 archivos de audio que fueron convertidos en formato *.Wav por su versatilidad de manejo con el software Matlab, cada uno de los 25 archivos tienen las características mostradas en la Tabla 1.

Cabe resaltar que se empleó una velocidad de muestreo de sonido de 11 Khz. Con la finalidad de cumplir con el criterio de Nyquist que es mayor ó igual a 2 veces la frecuencia de muestreo, que para nuestro caso pertenece al canal telefónico que es aproximadamente 4 Khz.

Tabla 1. Características de cada archivo de voz..

Velocidad de Transmisión.	128 Kbps.
Tamaño de muestra de sonido.	16 bits.
Tipo de canal.	Monofónico.
Velocidad de muestreo de sonido.	11 Khz.
Formato de audio.	*.Wav

4.2 Preprocesamiento

El objetivo de esta etapa es acondicionar la señal de entrada para que esta pueda ser procesada por la RNA, primero acotamos la señal de voz eliminando la parte inicial y final de la misma, que solo representan ruido para obtener la señal de voz a la cual le aplicaremos las Wavelets, como se ve en la Figura 4, se toma la subseñal *a[n]* correspondiente a las bajas frecuencias de la señal de voz donde se localiza la mayor cantidad de energía de la misma, despreciándose la subseñal *b[n]* que corresponde a las altas frecuencias ya que es donde se encuentra la mayor cantidad de ruido de la señal (ruido ambiental y el ruido del canal telefónico). Obteniendo así una señal de voz compacta y filtrada con respecto a la original. Posteriormente se normaliza la señal de voz resultante, para finalmente extraer los coeficientes LPC de la señal, que servirán para el diseño de los patrones de entrenamiento de la RNA.

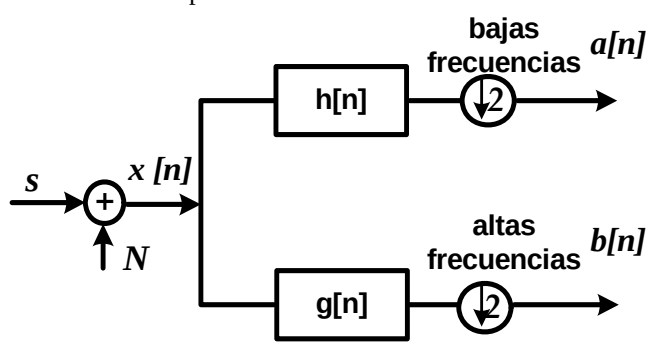

Fig. 4. Estructura de la Wavelet.

La etapa del preprocesamiento de la señal de voz consiste de los siguientes pasos, los cuales se observan en la Figura 2.

Acotamiento de la señal: En esta etapa se eliminan las muestras de tiempo que sólo contienen "silencios" diferentes a las características acústicas de cada hablante, por lo general estas se encuentran al principio y al final de los archivos de audio. El resultado de este proceso se muestra en la Figura 5.

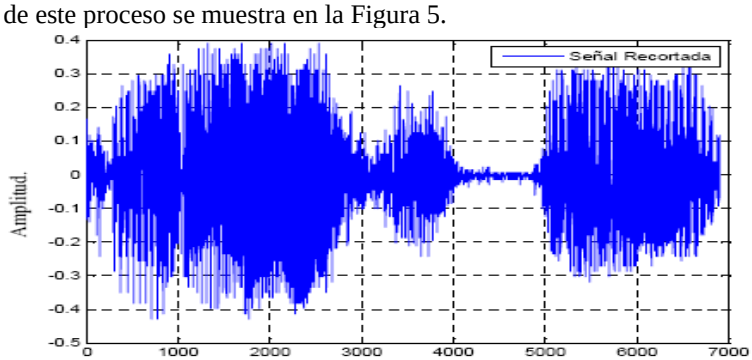

Fig. 5. Recorte de la señal de audio.

Wavelets: Toda señal de voz en la Naturaleza se encuentra afectada por ruido, y la señal de voz del canal telefónico no es la excepción. Por tal motivo se emplean las Wavelets para reducir este efecto. En este trabajo se propone emplear tres tipos de Wavelets las cuales son Haar, Coiflet y Daubechies. Observando que la mejor de ellas es la wavelet Daubechies debido a que, presenta el mayor porcentaje de compactación de energía de la subseñal $a[n]$ para cada uno de los veinticinco archivos. Esto permite eliminar la subseñal $b[n]$ de altas frecuencias.

Normalización: La normalización consiste en ajustar todos los parámetros a una sola escala para que al momento de ser utilizados por la RNA no causen problemas de estabilidad, en este caso la escala empleada se encuentra dada por los parámetros de la función de activación de la RNA que es una tangente bipolar sigmoidal y trabaja con valores de [-1,1], por lo tanto cada uno de los 25 archivos que previamente fueron compactados y filtrados por medio de las Wavelets es normalizado a esta escala, como se observa en (10), donde los datos que se quieren normalizar se encuentran dentro del vector $x[i]$, con i=$1,\ldots,n$. El procedimiento a seguir es el siguiente:

a) Se calcula la media (u) y la desviación estándar (σ) del vector $x[n]$.

b) Se normalizan los datos según la relación:

$$\dot{x}[n] = \frac{x[n]-\mu}{\sigma} \tag{10}$$

c) Se calculan el máximo y el mínimo del vector $\dot{x}[n]$, se divide por el de mayor valor absoluto y los datos normalizados caen dentro del intervalo $[-1,1]$.

Los resultados se ilustran en la Figura 6.

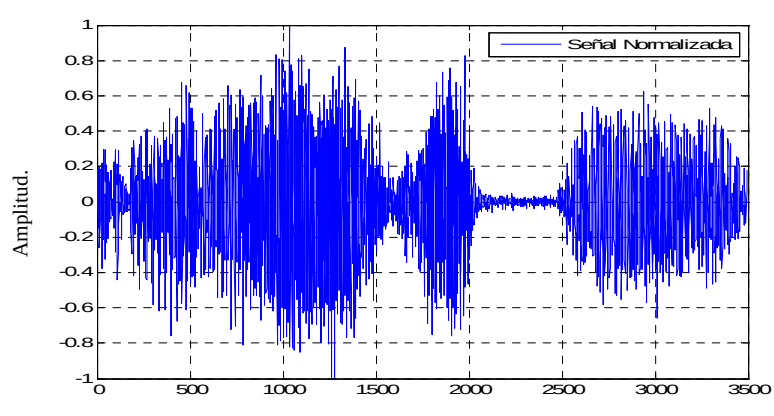

Fig. 6. Señal de audio normalizada.

Extracción de los coeficientes LPC: Debido a las propiedades mencionadas en la Sección 2.1 de los coeficientes LPC, en particular a que son capaces de modelar con gran aproximación la información lingüística y la zona vocal humana, en este trabajo se propone emplear diferentes números de coeficientes a_i descritos en (2) formando así una matriz de [$25\ x\ n$] elementos que corresponden a la extracción de n coeficientes para cada uno de los 25 archivos formados por los hablantes, creando de esta manera el patrón de entrenamiento de la RNA, por consiguiente se cambian el número de capas de entrada y oculta de la RNA con los siguientes resultados, como se puede ver en la Tabla 2.

Tabla 2. Coeficientes LPC.

COEFICIENTES LPC			
LPC	**Capa de Entrada**	**Capa Oculta**	**Efectividad**
2	3	8	60%
3	4	9	80%
4	5	10	100%
5	6	10	100%
6	7	10	100%
10	11	8	100%
15	16	8	100%

4.3 Diseño de la Red Neuronal Artificial

Esta etapa consiste en dos partes, la primera de ellas es el entrenamiento de la RNA, la cual se lleva a cabo con la finalidad de modificar los pesos de la red en cada una de las capas, de manera que coincida la salida deseada por el usuario con la salida obtenida por la red ante la presentación de un determinado patrón de entrada.

La segunda consiste en una fase de validación de la red frente a cualquier patrón de entrada que le sea presentado. Se empleó una arquitectura Backpropagation con tres capas, la capa de entrada, oculta y la de salida.

Fase de entrenamiento: Para el correcto desempeño de esta fase se emplearon, los valores establecidos para la capa de entrada y oculta mostrados en la Tabla 3 con los siguientes parámetros:

1) Neuronas de la capa de entrada=25.
2) Neuronas de la capa oculta=21.
3) Neuronas de la capa de salida=5.
4) Número de entrenamientos=25.
5) Número de épocas=700.
6) Pesos de la capa de entrada y de la capa oculta. (valores dentro de un rango de ([2.4 -2.4]) /Neuronas de entrada).[4]
7) Patrón de entrenamiento.
8) Salida deseada.
9) Error cuadrático medio requerido=0.005.
10) Tasa de aprendizaje =.009, .05, 0.02.

Bajo estos parámetros y basándonos en el punto 3 donde se explica detalladamente el funcionamiento de la RNA se entrenó a la misma, una vez entrenada se evalúa la RNA con el número de entrenamientos propuestos para generar y guardar los pesos de la capa oculta y de salida ya entrenados para emplearse en la próxima etapa.

Fase de evaluación: Se abren los pesos guardados obtenidos para la capa oculta y de salida del proceso de entrenamiento, se definen los puntos (1-4,6 y 7) de la fase de entrenamiento, se evalúa la red con un sólo patrón de entrenamiento el cual es el objetivo a identificar dentro de nuestra RNA, si el patrón de entrenamiento se encuentra la RNA lo identifica con uno de los posibles hablantes empleados en el entrenamiento de acuerdo a las características de los valores de sus pesos, sino se encuentra dentro de los hablantes empleados en el entrenamiento de la red se emite un mensaje de error indicando que la persona no ha podido ser identificada.

5 Resultados obtenidos

Tabla 3. Resultados obtenidos..

PRUEBAS CON DIFERENTES ESTADOS DE ANIMO					
	Alejandro	Leydi	Orlando	Diana	Luis
Hablantes	A1=Ide	Le1=Ide	O1=Ide	D1=Ide	Lb1=Ide
	A2=Ide	Le2=Ide	O2=Ide	D2=Ide	Lb2=Ide
	A3=Ide	Le3=Ide	O3=Ide	D3=Ide	Lb3=Ide
	A4=Ide	Le4=Ide	O4=Ide	D4=Ide	Lb4=Ide
	A5=Ide	Le5=Ide	O5=Ide	D5=Ide	Lb5=Ide
Recono-cimiento	100%	100%	100 %	100%	100%
				Efectividad=	100%

La Tabla 3 muestra los resultados obtenidos en esta investigación tomando en cuenta los valores propuestos en la fila 3 de la Tabla 2, con lo cual observamos que nuestros resultados son bastante ideales debido a que obtenemos una efectividad del 100%.

Continuando con nuestras pruebas, al momento de evaluar la RNA con los archivos de voz sin que estos hayan pasado por la etapa del preprocesamiento los resultados obtenidos en efectividad disminuyen del 100% al 96%.

Graficando las variaciones de la tasa de aprendizaje obtenemos diferentes valores de error para el proceso de entrenamiento de la RNA, que se muestran en la Figura 7. De la gráfica observamos que los mejores valores para la construcción de la RNA, son

los de la línea más gruesa (alfa=0.009,0.05, 0.02) ya que con ellos obtenemos los mínimos errores en la RNA.

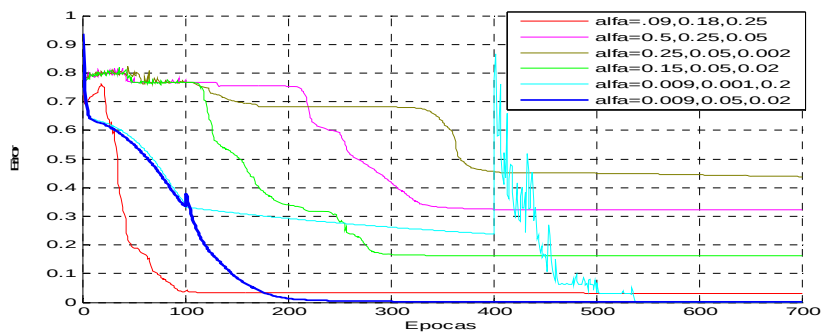

Fig. 7. Gráfica del error.

6 Conclusiones

Con el sistema propuesto en este trabajo se tiene un buen funcionamiento para esta aplicación ya que obtenemos un reconocimiento de los hablantes del 100% como se da a notar en la tabla de resultados.

Cabe destacar que el procedimiento para la obtención de los valores de los parámetros empleados en el diseño de la RNA Backpropagation no existen como tales bien definidos, sin embargo, los valores propuestos en este trabajo fueron obtenidos a prueba y error dándonos cuenta de cómo la tasa de aprendizaje y la elección correcta de los pesos iniciales influye mucho en el resultado obtenido.

Podemos observar que con sólo 4 coeficientes es suficiente para aproximar correctamente una señal de voz. El sistema propuesto presenta una estructura fácil de desarrollar y su complejidad matemática es mínima, por lo que puede tener diversas aplicaciones en el campo de la identificación y verificación de hablantes.

Referencias

1. José R Hilera Martínez, “Redes Neuronales Artificiales, Fundamentos, Modelos y Aplicaciones”, Alfa Omega, México, (2000)
2. Simon Haykin, “Neural Networks”, Prentice - Hall, New Jersey, (1999)
3. Sadaoki Furui, “Digital Speech Processing, Synthesis, and Recognition, Cambridge University Press, (2001)
4. Bo nifacio Martín del Rio, Alfredo Sanz Molina. "Redes Neuronales y Sistemas Borrosos", Ra-Ma, Madrid, (2001)
5. Laurene Fausett. “Fundamentals Neuronal Network, architectures, algorithms, and applications”, Prentice – Hall, New Jersey, (1995).
6. Jose Luis Oropeza. “Algorithms and Methods for the Automatic Speech Recognition in Spanish Languages using Syllables”, Computación y Sistemas, Vol.9, No. 3, pp.270-286, 2006.