Seminario de Tesis II
Redes LSTM para el reconocimiento de voz aplicado a un conjunto de dígitos

Victor Jesus Sotelo Chico

Universidad Nacional de Ingeniería
Facultad de Ciencias
Escuela Profesional de Ciencias de la Computación
Asesor: Dr. Antonio Morán Cardenas

17 Diciembre 2018

Contenido

Introducción

Las señales de voz proveen una gran cantidad de información a través del tiempo, el estudio de éstas ha permitido el desarrollo de sistemas de reconocimiento de voz.

Objetivos

- Conocer el proceso involucrado en el habla humana.
- Estudiar procesamiento de las señales de voz.
- Conocer las ventajas del uso de redes recurrentes.
- Diseñar una red neuronal capaz de reconocer un conjunto de audios de números.
- Mostrar los resultados obtenidos y explicarlos basándonos en la teoría estudiada.

Estado del arte

- Análisis de los algoritmos y aplicaciones en el reconocimiento de voz.
- Reconocimiento de voz con redes recurrentes profundas.
- Reconocimiento de voz usando las técnicas Mel Frequency Cepstral Coefficient (MFCC) y Dynamic Time Warping (DTW).
- Redes convolucionales para reconocimiento de voz.

Redes neuronales recurrentes(RNN)

Este tipo de redes son utilizadas principalmente para tratar con una información de secuencia. Sin embargo presentan 2 problemas:

- Problema de desaparición de gradiente.
- Problema de las dependencias a largo plazo.

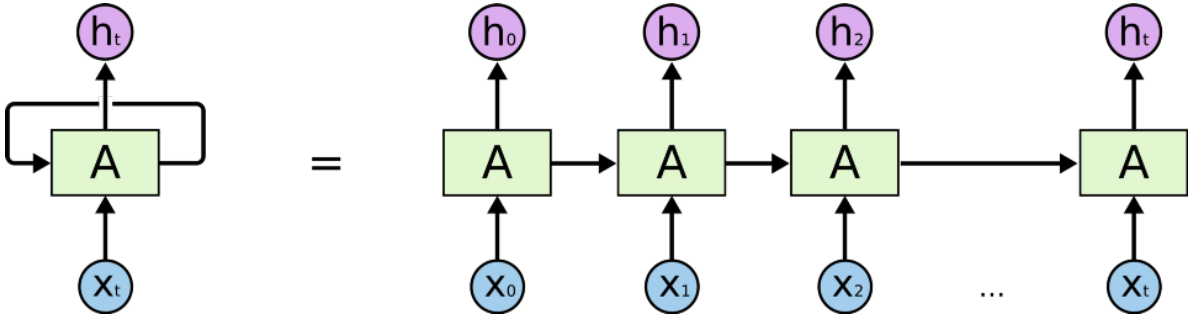


Figura: Versión desenrollada - Fuente: *http://colah.github.io*

Redes LSTM(Long Short Term Memory)

Son un tipo de RNN que se encargan de transmitir la información de los lapsos de tiempo y recordarla.

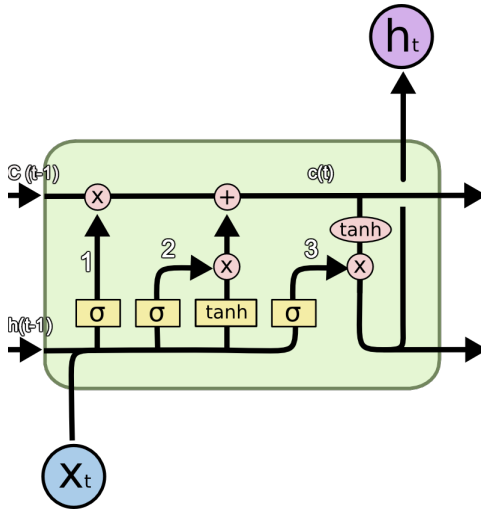


Figura: Esquema de una unidad LSTM - Fuente: *http://colah.github.io*

- X_t: entrada actual
- σ, *tanh*: capas
- h_t: salida en t .
- C_t: memoria en t

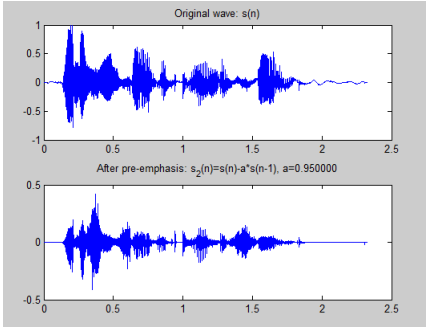
- Forget Layer
  $f_t = \sigma(W_t \cdot [h_{t-1}, x_t] + b_f)$
- Input Layer
  $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$
- Vector de valores posibles
  $\hat{C}_t = \sigma(W_C \cdot [h_{t-1}, x_t] + b_C)$
- Actualización de celda de estado
  $C_t = f_t * C_{t-1} + i_t * \hat{C}_t$
- Unidades ocultas
  $o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$
  $h_t = o_t * \tanh(C_t)$

Mel Frequency Cepstral Coefficients (MFCC)

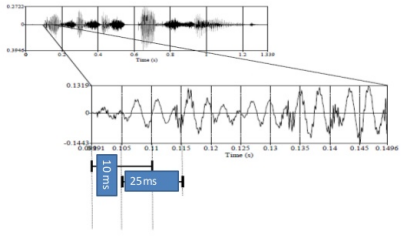
Es el algoritmo más usado en la extracción de características de una señal de voz. Posee las siguientes etapas:

- Pre-emphasis
- Framing
- Hamming windowing
- Transformada rápida de Fourier (FFT)
- Mel filter bank
- Discrete Cosine Transform(DCT)

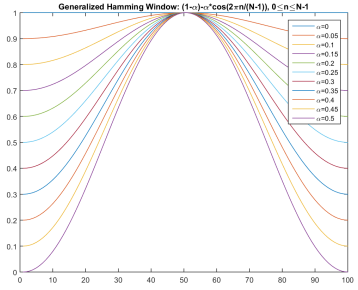
MFCC



(a) Pre-emphasis
Fuente:*www.researchgate.net*

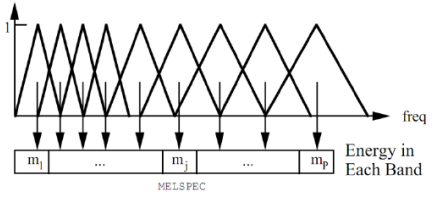


(b) Framming
Fuente:*www.slideshare.net*



(c) Hamming windowing

Fuente:*www.slideshare.net*



(d) Mel filter bank
Fuente:
https://www.researchgate.net

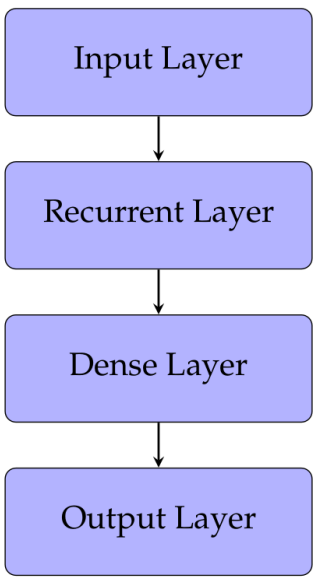
Datos

Para este trabajo se recolecto muestras de hablantes 7 varones y 5 mujeres.

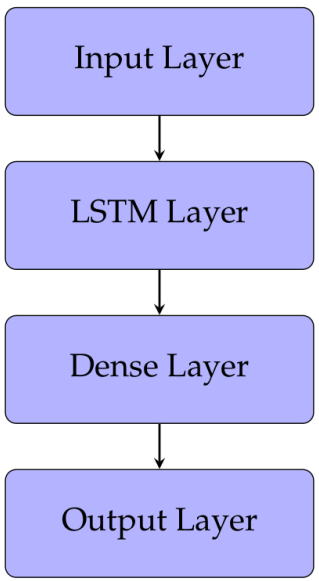
conjunto de datos			
Training		Test	
VF	VM	VF	VM
90	150	40	70

Figura: División del conjunto en base a la cantidad de audios con voces femeninas y masculinas - Fuente:*Elaboración propia*

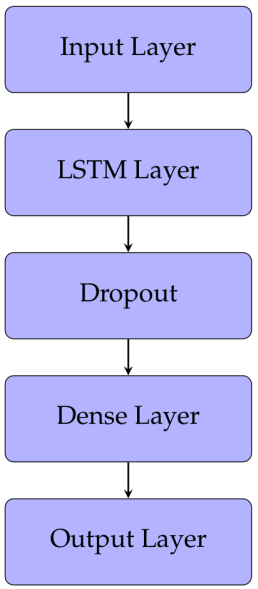
Esquema de pruebas



(a) RNN simple



(b) LSTM simple



(c) LSTM con dropout

Figura: Esquemas de prueba - Fuente:*Elaboración propia*

Esquema Propuesto

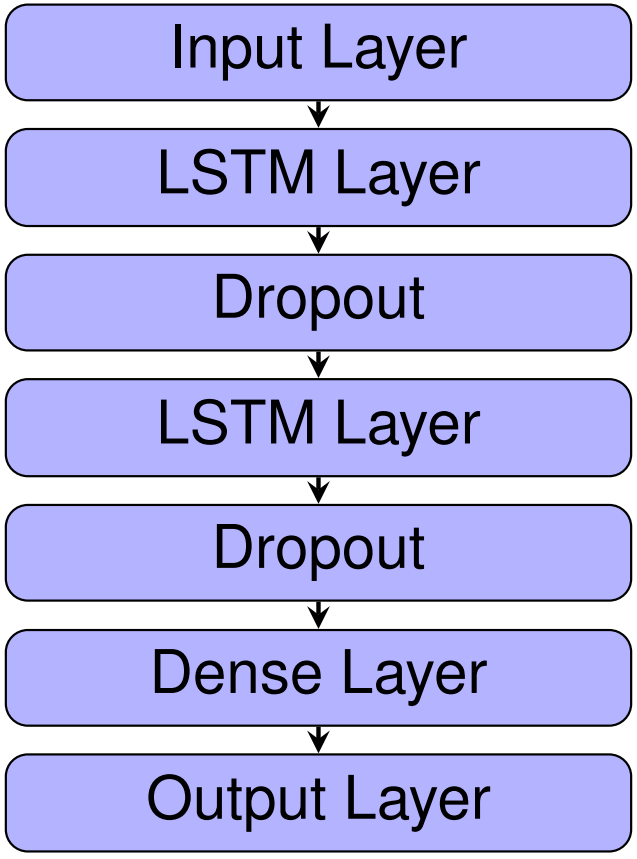
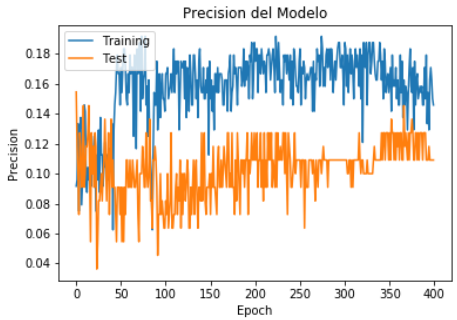


Figura: Esquema propuesto - Fuente: *Elaboración propia*

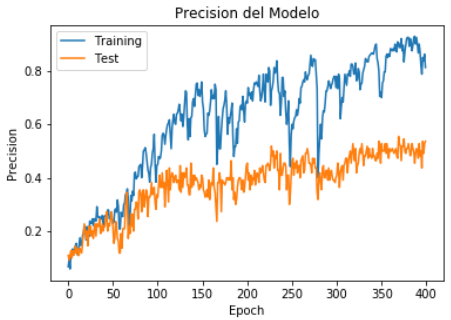
Librerías utilizadas

- **LibROSA**: Librería usada para la extracción de características MFCC.
- **Tensorflow**: Usado para la distribución del entrenamiento en la GPU.
- **Keras**: Utilizado para el diseño de las redes recurrentes.
- **Sklearn**: Aplica one hot encoding a nuestras variables categóricas.

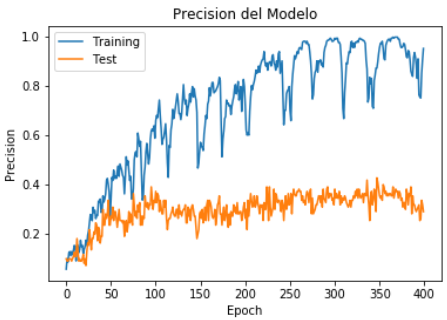
Resultados 64 unidades ocultas



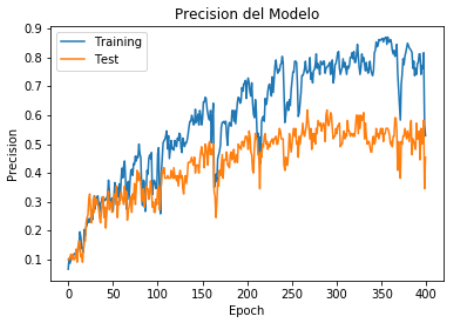
(a) RNN simple



(b) LSTM simple



(c) LSTM dropout 0.5



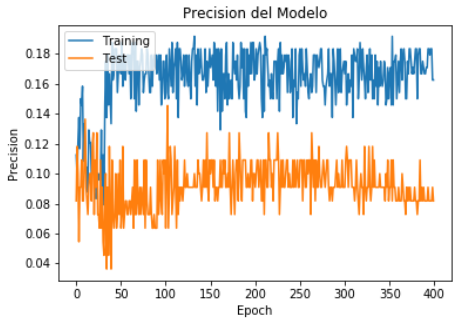
(d) LSTM dropout 0.8

Figura: Resultados 64 unidades ocultas - Fuente:*Elaboración propia*

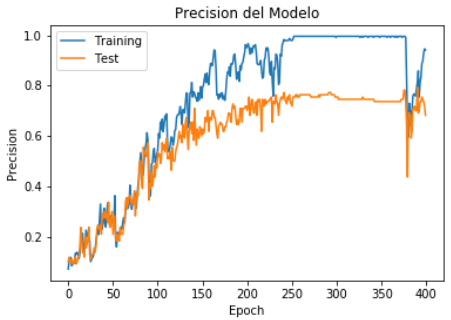
Modelo	Precisión test(%)	Precisión Training (%)
RNN	10.90	14.58
LSTM	40.00	73.75
LSTM Dropout(0.5)	31.81	98.33
LSTM Dropout(0.8)	55.00	77.91

Figura: Resultados 64 unidades ocultas 300 iteraciones - Fuente:*Elaboración propia*

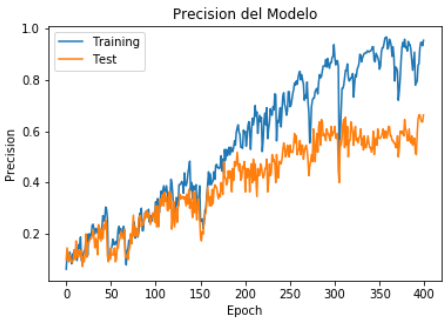
Resultados 128 unidades ocultas



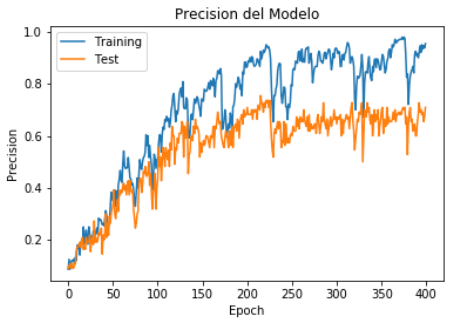
(a) RNN simple



(b) LSTM simple



(c) LSTM dropout 0.5



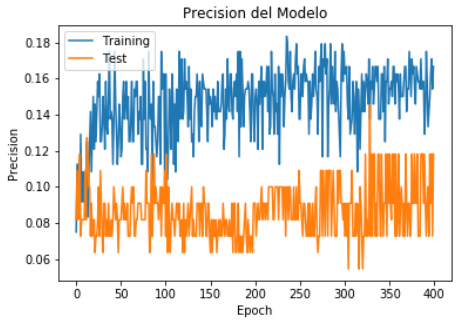
(d) LSTM dropout 0.8

Figura: Resultados 128 unidades ocultas - Fuente:*Elaboración propia*

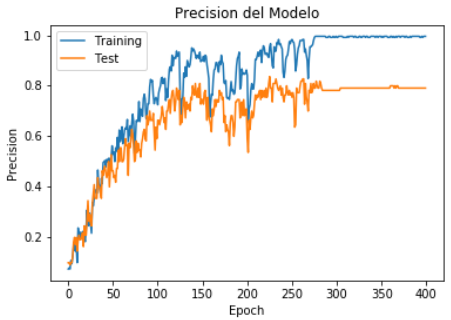
Modelo	Precisión test(%)	Precisión Training (%)
RNN	9.09	17.50
LSTM	75.45	99.16
LSTM Dropout(0.5)	60.00	93.74
LSTM Dropout(0.8)	67.27	93.74

Figura: Resultados 128 unidades ocultas 300 iteraciones - Fuente:*Elaboración propia*

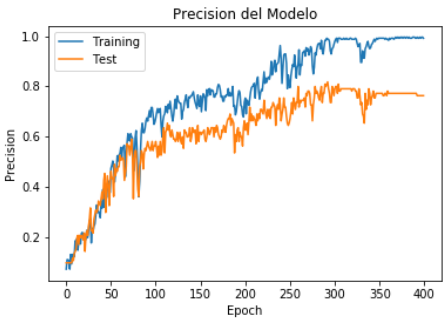
Resultados 256 unidades ocultas



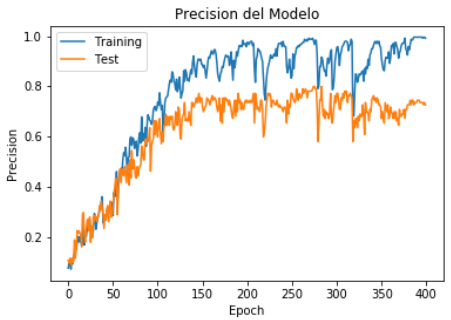
(a) RNN simple



(b) LSTM simple



(c) LSTM dropout 0.5



(d) LSTM dropout 0.8

Figura: Resultados 256 unidades ocultas - Fuente:*Elaboración propia*

Modelo	Precisión test(%)	Precisión Training (%)
RNN	9.09	17.50
LSTM	78.18	99.58
LSTM Dropout(0.5)	76.36	96.66
LSTM Dropout(0.8)	82.57	92.91

Figura: Resultados 256 unidades ocultas con 300 iteraciones - Fuente:*Elaboración propia*

Resultados del modelo propuesto

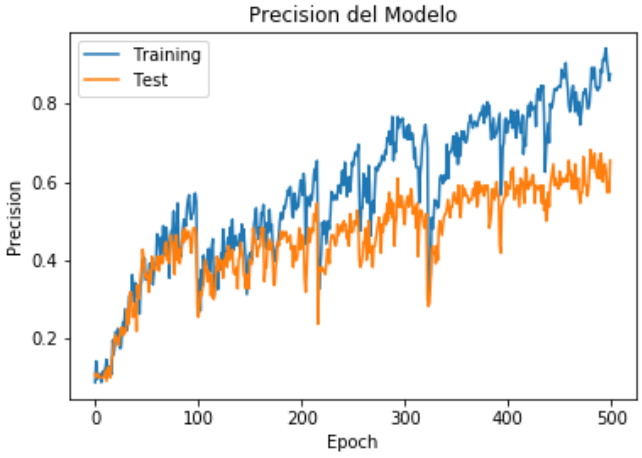


Figura: Precisión 64 unidades ocultas - Fuente:*Elaboración propia*

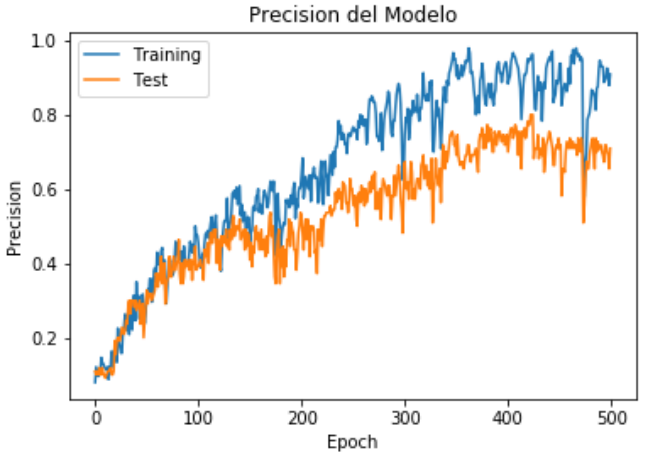


Figura: Precisión 128 unidades ocultas - Fuente:*Elaboración propia*

Tabla de resultados

Modelo 2LSTM 2 Dropout	Precisión test(%)	Precisión Training (%)
64 estados ocultos	65.45	87.50
128 estados ocultos	70.91	90.83

Figura: Resultados de precisión 2 capas LSTM con 500 iteraciones - Fuente:*Elaboración propia*

Conclusiones

- El uso 2 capas LSTM y dropout de 0.4 mejoran el rendimiento de nuestro modelo además manejar mejor el problema de overfitting.
- Las redes LSTM obtuvieron mejores resultados contra el RNN simple debido a su capacidad de manejar el problema de desaparición de gradiente.
- El número de estados ocultos puede acelerar el proceso de entrenamiento pero puede generar efectos de overfitting.

Conclusiones

- El algoritmo MFCC permite extraer las características necesarias para el procesamiento de la señal de voz.
- Los audios con distintos tiempos incrementan el número de lapsos tiempos en que desenrolla la red lo cual impide una mejor precisión.
- La cantidad de epochs requeridos es menor en las redes con una sola capa LSTM pero estas tienden a caer en problemas de overfitting.

Trabajos Futuro

En futuros trabajos se tratara de construir sistema de reconocimiento para más palabras además de permitir reconocer un lenguaje básico de modo que la red reconozca oraciones completas.