

# Práctica Final: EMNIST - Clasificación de imágenes:



# UNIVERSIDAD DE GRANADA

Luis González Romero, XXXXXXXXXX, luisgonromero@correo.ugr.es

Escuela Técnica Superior de Ingeniería informática y Telecomunicaciones

3 de febrero de 2021

# Práctica Final: EMNIST - Clasificación de imágenes

Memoria sobre la última práctica de la asignatura Inteligencia de Negocio  
cursada en la ETSIIT, UGR.

Luis González Romero, XXXXXXXXXX, luisgonromero@correo.ugr.es

## Índice

1. Descripción y análisis del problema	6
2. Descripción de algoritmos y estudio experimental	7
3. Planteamiento de futuro, ¿qué harías si puedes trabajar durante 6 meses en el problema?	9

## Índice de figuras

1.	Proceso de conversión del dataset. . . . .	6
2.	Capas de los modelos . . . . .	7

## Índice de tablas

1.	Estructura y organización del dataset EMNIST . . . . .	6
2.	Configuraciones disponibles con <i>tensorflow_datasets</i> . . . . .	6
3.	Accuracy obtenida en test para los distintos modelos . . . . .	8
4.	Loss obtenida en test para los distintos modelos . . . . .	8

## 1. Descripción y análisis del problema

EMNIST(Extended MNIST) trata un problema de clasificación de imágenes, en este caso de un conjunto de caracteres escritos a mano que mantienen la estructura de imagen del original MNIST.

Al usar la misma estructura que MNIST, se mantiene la compatibilidad con distintos sistemas y modelos que existen para el conjunto de datos original:

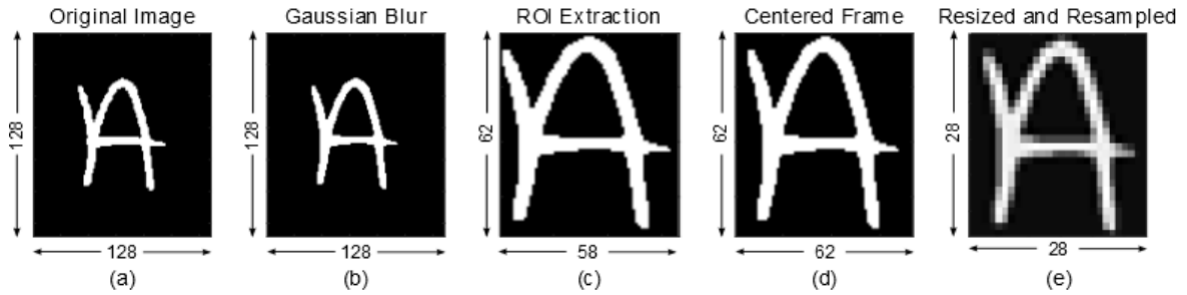


Figura 1: Proceso de conversión del dataset.

Para este conjunto de datos, existen varias particiones disponibles de cara a tener mejor definida la clasificación correspondiente y la estructura del dataset:

Name	Classes	No. Training	No. Testing	Total
<i>Complete</i>	62	697,932	116,323	814,255
<i>Merge</i>	47	697,932	116,323	814,255
<i>Balanced</i>	47	112,800	18,800	131,600
<i>Digits</i>	10	240,000	40,000	280,000
<i>Letters</i>	37	88,800	14,800	103,600
<i>MNIST</i>	10	60,000	10,000	70,000

Tabla 1: Estructura y organización del dataset EMNIST

En este dataset a diferencia del clásico MNIST, se añaden dificultades

Para hacer uso de este dataset he seguido la recomendación de Nuria con el catálogo de datasets ofertados por *tensorflow\_datasets*.

	No. muestras	No. Clases	¿Clases balanceadas?
<i>EMNIST ByClass</i>	814,255	62	No
<i>EMNIST ByMerge</i>	814,255	47	No
<i>EMNIST Balanced</i>	131,600	47	Sí
<i>EMNIST Letters</i>	145,600	26	Sí

Tabla 2: Configuraciones disponibles con *tensorflow\_datasets*

## 2. Descripción de algoritmos y estudio experimental

Empecé usando el dataset *Letters* ya que al estar balanceado y tener menos clases puede ser mas fácil aunque debe clasificar con la misma etiqueta a un mismo carácter(letra) para su mayúscula y su minúscula. Usando modelos de keras que encontré en blogs como primera toma de contacto llegué a obtener un accuracy en torno a 0.75.

Tras esto, pasé a buscar modelos en Kaggle algo más complejos como nos sugirió Nuria. De los que probé, me quedé con los siguientes dos que ya empezaron a dar accuracy cercanas a 0.9 aunque como se puede ver, con un alto valor de loss. Y como ultimo modelo, usé como referencia el blog que nos recomendó Nuria para reproducir las capas.

En todos los modelos usé un 20 % del conjunto de entrenamiento para validación.

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 26, 26, 32)	320
max_pooling2d (MaxPooling2D)	(None, 13, 13, 32)	0
flatten (Flatten)	(None, 5408)	0
dense (Dense)	(None, 512)	2769408
dense_1 (Dense)	(None, 128)	65664
dense_2 (Dense)	(None, 26)	3354
Total params: 2,838,746		
Trainable params: 2,838,746		
Non-trainable params: 0		

(a) CNN(Kaggle)

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 28, 28, 1)]	0
conv2d (Conv2D)	(None, 28, 28, 32)	832
max_pooling2d (MaxPooling2D)	(None, 14, 14, 32)	0
conv2d_1 (Conv2D)	(None, 14, 14, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 7, 7, 64)	0
conv2d_2 (Conv2D)	(None, 7, 7, 128)	73856
max_pooling2d_2 (MaxPooling2D)	(None, 3, 3, 128)	0
flatten (Flatten)	(None, 1152)	0
dense (Dense)	(None, 256)	295168
dense_1 (Dense)	(None, 128)	32896
dense_2 (Dense)	(None, 26)	3354
Total params: 424,602		
Trainable params: 424,602		
Non-trainable params: 0		

(b) CNN con 2 capas(Kaggle)

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 26, 26, 32)	320
max_pooling2d (MaxPooling2D)	(None, 13, 13, 32)	0
dropout (Dropout)	(None, 13, 13, 32)	0
conv2d_1 (Conv2D)	(None, 11, 11, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 5, 5, 64)	0
dropout_1 (Dropout)	(None, 5, 5, 64)	0
flatten (Flatten)	(None, 1600)	0
dense (Dense)	(None, 512)	819712
dropout_2 (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 26)	13338
Total params: 851,866		
Trainable params: 851,866		
Non-trainable params: 0		

(c) CNN con 2 capas(Blog)

Figura 2: Capas de los modelos

	Letters	Balanced	ByMerge
<i>Primer modelo</i>	0.9070	-	-
<i>Segundo modelo</i>	0.9213	0.8786	-
<i>Tercer modelo</i>	0.9293	0.8873	0.8999

Tabla 3: Accuracy obtenida en test para los distintos modelos

	Letters	Balanced	ByMerge
<i>Primer modelo</i>	0.5046	-	-
<i>Segundo modelo</i>	0.2308	0.3451	-
<i>Tercer modelo</i>	0.2063	0.3275	0.2716

Tabla 4: Loss obtenida en test para los distintos modelos

Tras ver que los modelos CNN funcionaban bien, pasé a ejecutar en el resto de datasets el que mejor funcionó para ver cómo se comportaba. Obtuvo buenos resultados para lo complejo que es el penúltimo dataset(*ByMerge*). Lo poco que pude mejorar el valor de loss fue ajustando el Dropout Regularization, pero poco más.

Intenté reproducir un ensemble de vgg, resnets y algún modelo más, pero no conseguí hacerlo ya que no estaba disponible el código y en el mío solamente encontraba errores. No me dió tiempo a hacerlo funcionar, así que terminé dejándolo fuera.



### **3. Planteamiento de futuro, ¿qué harías si puedes trabajar durante 6 meses en el problema?**

De cara a dedicarme por completo al problema, me gustaría estudiar modelos de la literatura para obtener una clasificación más sólida en el dataset *Letters* antes de pasar al resto.

Por ejemplo, me gustaría estudiar la hibridación de algoritmos evolutivos con las redes convolucionales sobre la que leí en este paper. Ya vimos en la asignatura de Metaheurísticas lo buenos resultados que se pueden obtener haciendo hibridaciones acertadas con los algoritmos evolutivos y explotar el uso de poblaciones de individuos, por lo que me interesaría mucho enfocarme en estudiarlo y ver su potencial.

Además, al ser esta mi primera asignatura de machine learning tendría que profundizar más en muchos aspectos, ya que a la hora de ajuste de parámetros por ahora siempre he tenido que leer mucho al respecto antes de realizarlo.