

Classificació d'imatges CNN amb Keras

Aquest projecte consisteix en la classificació d'imatges mitjançant una xarxa neuronal convolucional (CNN) utilitzant la llibreria Keras. S'ha utilitzat el dataset "Places" que conté imatges de diferents llocs al voltant del món. Aquest dataset es va obtenir de Kaggle.

Descripció de les llibreries a utilitzar

Aquest projecte fa servir les següents llibreries:

- NumPy: Llibreria d'àlgebra lineal per a Python que proporciona estructures de dades eficients per a operacions numèriques i científiques en general.
- Pandas: Llibreria de manipulació de dades que ofereix estructures de dades de gran potència i flexibilitat per a la gestió de dades en forma de taules.
- Matplotlib: Llibreria de visualització de dades que permet crear gràfics i figures de gran qualitat per a la representació visual de dades.
- Scikit-learn: Llibreria de machine learning de codi obert que ofereix algoritmes per a tasques de classificació, regressió i agrupació de dades, així com eines per a la preparació de dades i la validació de models.
- Seaborn: Llibreria de visualització de dades que permet crear gràfics estadístics atractius i informatius, especialment dissenyats per a l'anàlisi de dades complexes i multidimensionals.
- Keras: Llibreria d'aprenentatge profund (deep learning) de codi obert escrita en Python que proporciona una interfície d'alta nivell per a la construcció de models de xarxes neuronals. La seva funcionalitat principal és permetre la creació, entrenament i avaluació de models de xarxes neuronals amb relativa facilitat, sense la necessitat d'escriure molts codi de baix nivell, el que fa que sigui una eina popular en el camp de l'aprenentatge profund.
- TensorFlow: Llibreria d'aprenentatge profund de codi obert desenvolupada per Google que permet la creació, entrenament i avaluació de models de xarxes neuronals. La seva funcionalitat principal és processar i manipular dades en forma de tensors (que són estructures multidimensionals), i utilitzar-los per construir models de xarxes neuronals per a tasques de classificació, regressió i altres problemes de machine learning. TensorFlow és una de les llibreries d'aprenentatge profund més populars i utilitzades actualment.

Estructura del projecte

Preparació de les dades

Les imatges originals es troben a la carpeta "places", que conté un 10% del total de 123.634 imatges. En aquest projecte s'ha utilitzat un dataset reduït, "places_reduced", que conté 23 carpetes de la "a" a la "w". Dins de cada carpeta hi ha un nombre de carpetes amb el nom d'una etiqueta i en aquestes carpetes és on es troben les imatges.

S'han seleccionat només les imatges de la carpeta "a" per començar a classificar. Aquestes imatges s'han dividit en tres conjunts: train, test i validació. La proporció d'aquesta divisió és del 70%, 20% i 10%, respectivament. En total, hi ha 6380 imatges a la carpeta de train, 1821 a la de test i 924 a la de validació.

Metodologia

En aquest projecte s'han utilitzat imatges de 256x256 píxels en format .jpg. S'ha preprocessat les imatges i s'han normalitzat a un rang de (0,1) mitjançant la funció de Keras. Això permet ser més robust i estable. També s'ha modificat el tamany de totes les imatges a 118x118 píxels amb 3 canals (RGB) utilitzant una funció de Keras.

S'ha utilitzat l'ImageGenerator de Keras per dividir les imatges en tres conjunts: train, test i validació. El conjunt de train conté 6380 imatges, test conté 1821 imatges i validació conté 924 imatges, amb 14 classes cada una.

S'han creat diferents models de xarxes neuronals convolucionals (CNN) per aconseguir la millor precisió en el problema de classificació d'imatges. Després d'haver entrenat el model de xarxa neuronal convolucional, s'ha fet un ajust final del model. Després d'aquest ajust, s'ha calculat el percentatge de precisió del model utilitzant les dades de test per avaluar la precisió del model en imatges que no ha vist abans.

S'ha creat una gràfica del model per visualitzar l'exactitud (precisió) i la pèrdua (loss) en el temps. Això ajuda a identificar si el model està sobreajustant (overfitting) o no.

s'ha fet una predicció

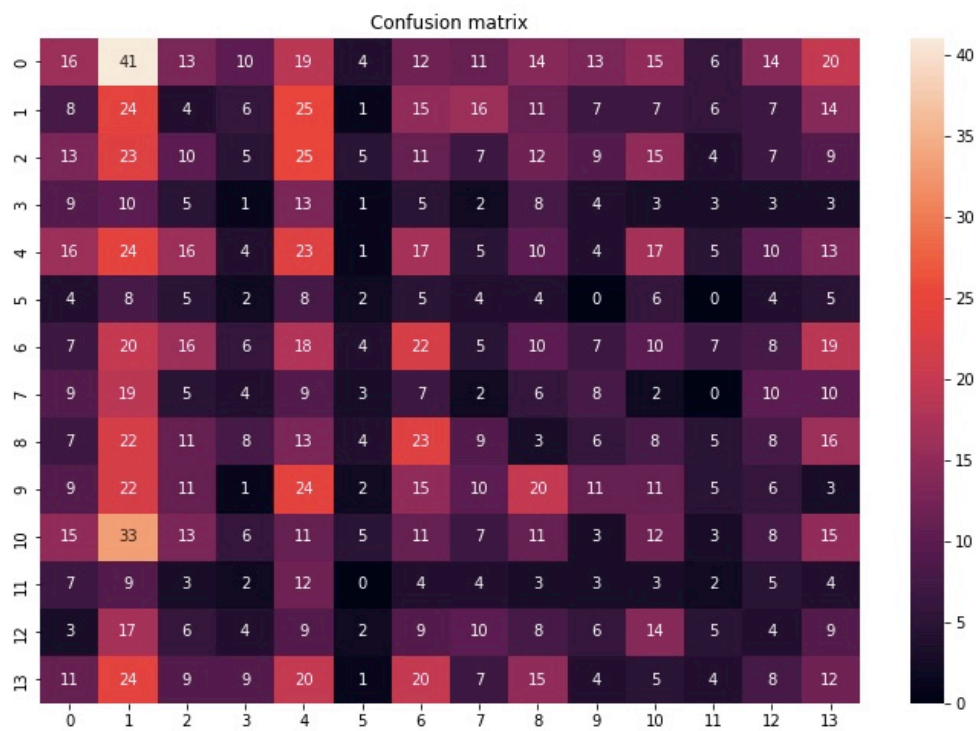
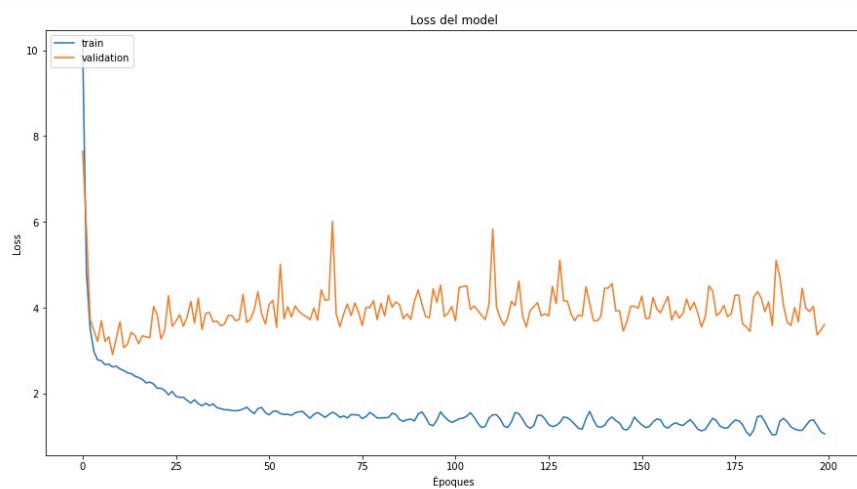
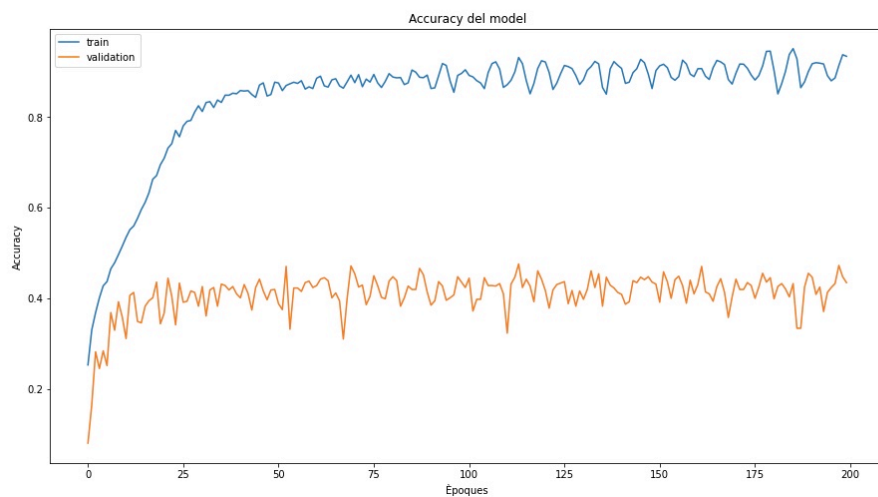
Finalment, s'ha creat una matriu de confusió per avaluar el rendiment del model en la classificació de cada classe. facilitat i quines són les que presenten més dificultats.

Model: "sequential_2"

Layer (type)	Output Shape	Param #
conv2d_4 (Conv2D)	(None, 126, 126, 32)	896
batch_normalization_2 (Batch Normalization)	(None, 126, 126, 32)	128
max_pooling2d_2 (MaxPooling2D)	(None, 63, 63, 32)	0
conv2d_5 (Conv2D)	(None, 61, 61, 64)	18496
batch_normalization_3 (Batch Normalization)	(None, 61, 61, 64)	256
max_pooling2d_3 (MaxPooling2D)	(None, 30, 30, 64)	0
conv2d_6 (Conv2D)	(None, 28, 28, 128)	73856
batch_normalization_4 (Batch Normalization)	(None, 28, 28, 128)	512
max_pooling2d_4 (MaxPooling2D)	(None, 14, 14, 128)	0
conv2d_7 (Conv2D)	(None, 12, 12, 128)	147584
batch_normalization_5 (Batch Normalization)	(None, 12, 12, 128)	512
max_pooling2d_5 (MaxPooling2D)	(None, 6, 6, 128)	0
flatten (Flatten)	(None, 4608)	0
dense (Dense)	(None, 512)	2359808
dropout (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 14)	7182
Total params: 2,609,230		
Trainable params: 2,608,526		
Non-trainable params: 704		

Resultats

4 capesConv, L2 Kernel regularizer, BatchNormalizer, Dropout=0,2, l'he entrenat 200 epochs.



El resultat ha sigut:
Test loss: 3.7795
accuracy: 0.4465

Conclusions

El resultat no ha sigut favorable, s'han de revisar els paràmetres per millorar l'entrenament, un altra cosa que podria ajudar seria augmentar les dades de "a" per millora l'entrenament i els resultats.

Adjunto una tabla amb els diferents paramentres utilitzats i resultats.

	model	num_conv_layers	batchnorm	dropout	L2	epochs	test_loss	test_accuracy
0	model26.h5	4	False	-	L2	100	3.606509	0.332784
1	modell27.h5	4	False	0.2	L2	100	2.033662	0.423943
2	model30.h5	4	True	0.2	L2	200	3.779453	0.446458
3	model01.h5	8	True	0.5	-	70	1.907737	0.425041

Referències:

<https://keras.io>

<https://www.tensorflow.org>

<https://www.kaggle.com>

<http://places.csail.mit.edu/user/>