



Implementación de una red neuronal convolucional para predecir si una araña posee importancia médica basado en respuestas de una cuenta de Twitter

José Alejandro López Quel

Universidad Galileo



Introducción

Se presenta un caso de predicción de la importancia médica de una araña basada en las respuestas de un aracnólogo en su cuenta de Twitter de imágenes de arañas compartidas. Se extrae los datos de la cuenta de Twitter y se etiqueta cada una de ellas para realizar entrenar una red neuronal convolucional. Por lo que se analiza en este trabajo el mejor método para realizar dicha clasificación.

Obtención de datos

Para obtener los datos se emplea la librería de python llamada Tweepy la cual emplea el API de Twitter para leer los datos de la plataforma, en este caso se analiza la cuenta de @Arachno_Cosas. Se lee las respuestas con mención que realiza el autor de la cuenta que contienen una imagen, por lo que se extrae el texto de la respuesta brindada y la imagen de la interacción.

Soy aracnólogo, pregúntame lo que quieras @Arachno... · Sep 21
¡Hola, @micaanabitate! Gracias por compartir. Perteneces a la familia #Lycosidae probablemente. No son consideradas de importancia médica #NIM Se les conoce como arañas lobo. Saludos

Mica Anabitate @micaanabitate · Aug 12
Hola @Arachno_Cosas Me encontré con esta amiga en el living de casa. Vivo en Australia, en NSW. Podrás decirme si es de importancia médica? Muchas gracias! (Ya fue liberada en el jardín)



Metodología

Para esta investigación se utiliza una red neuronal compuesta por seis capas intermedias, las cuales se van alternando entre convolución 2D y max pooling 2D, utilizando como función de activación ReLU y reduciendo el número de unidades a utilizar iniciando en 64 unidades hasta reducirse a 16. Luego de ello se agrega una capa de dropout con el 45% de los datos para después emplear la capa de Flatten y por ultimo una capa densa de 128 unidades con activación ReLU para finalizar con una función sigmoideal.

| Model: "sequential_1" | | |
|-------------------------------|----------------------|---------|
| Layer (type) | Output Shape | Param # |
| ----- | | |
| conv2d_3 (Conv2D) | (None, 222, 222, 64) | 1792 |
| max_pooling2d_3 (MaxPooling2) | (None, 111, 111, 64) | 0 |
| conv2d_4 (Conv2D) | (None, 109, 109, 32) | 18464 |
| max_pooling2d_4 (MaxPooling2) | (None, 54, 54, 32) | 0 |
| conv2d_5 (Conv2D) | (None, 52, 52, 16) | 4624 |
| max_pooling2d_5 (MaxPooling2) | (None, 26, 26, 16) | 0 |
| dropout_1 (Dropout) | (None, 26, 26, 16) | 0 |
| flatten_1 (Flatten) | (None, 10816) | 0 |
| dense_2 (Dense) | (None, 128) | 1384576 |
| dense_3 (Dense) | (None, 2) | 258 |
| ----- | | |
| Total params: 1,409,714 | | |
| Trainable params: 1,409,714 | | |
| Non-trainable params: 0 | | |

```
4/4 [=====] - 12s 3s/step - loss: 4.0485 - accuracy: 0.4583 - val_loss: 0.6722 - val_accuracy: 0.9000
Epoch 2/10
4/4 [=====] - 10s 2s/step - loss: 0.6198 - accuracy: 0.7833 - val_loss: 0.6317 - val_accuracy: 0.9000
Epoch 3/10
4/4 [=====] - 10s 2s/step - loss: 0.6177 - accuracy: 0.7833 - val_loss: 0.6353 - val_accuracy: 0.9000
Epoch 4/10
4/4 [=====] - 10s 2s/step - loss: 0.6566 - accuracy: 0.7833 - val_loss: 0.6038 - val_accuracy: 0.9000
```

Resultados

Se obtiene que los dos experimentos realizados con el modelo planteado obtienen los mismos resultados de accuracy de 0.9. Por lo que el modelo planteado cumple con identificar los valores de Importancia Médica para las imagines seleccionadas como validación.

Se obtienen buenos resultados dado a que se generan variaciones de la misma imagen para el entrenamiento empleando Image Data Generator y los parámetros:

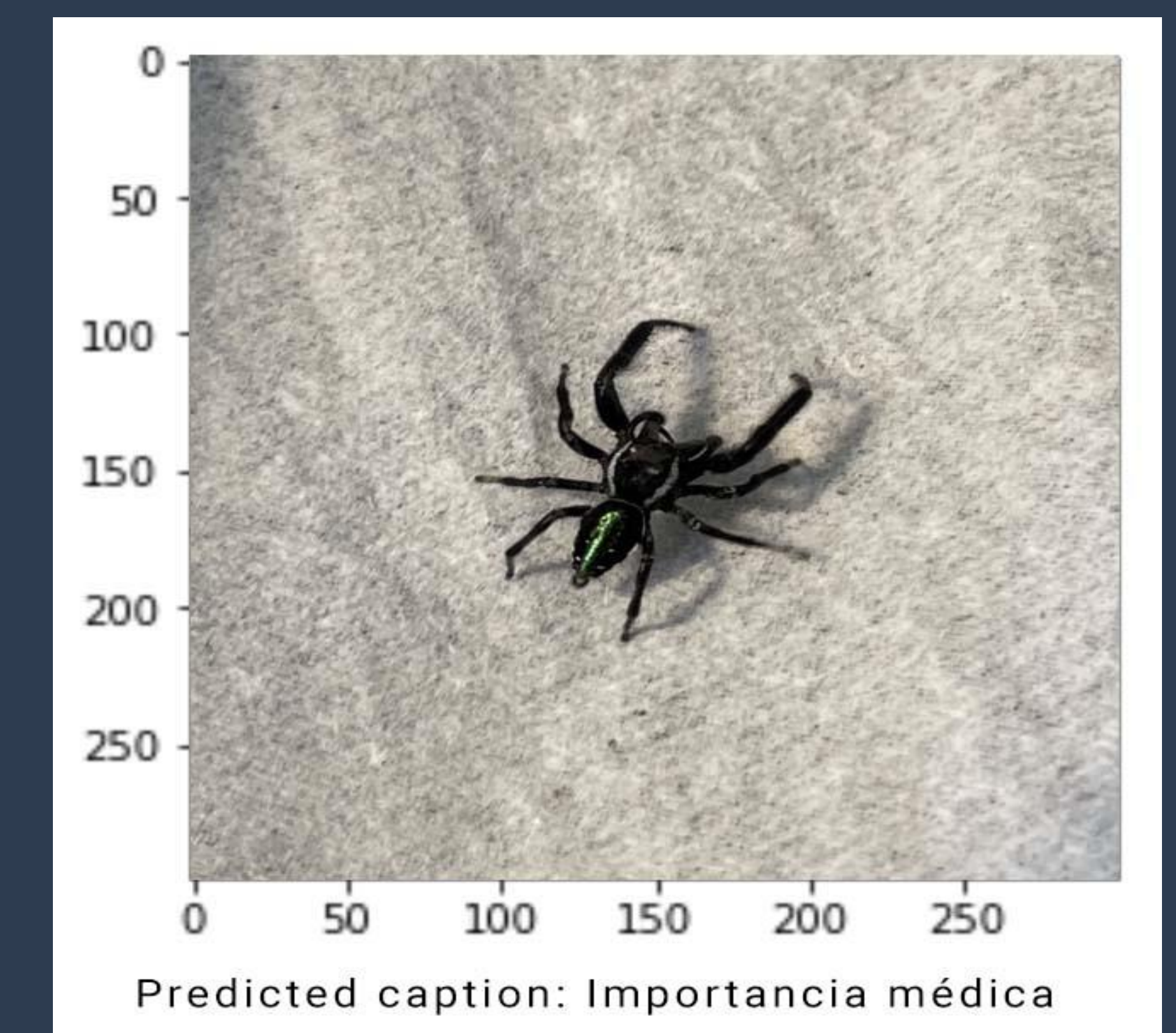
- rotation_range
- zoom_range
- width_shift_range
- height_shift_range
- horizontal_flip



Conclusiones

Se comprueba que al utilizar la función de ImageDataGenerator de Keras se pueden llegar a generar más variaciones de las imágenes lo cual ayuda al modelo a tener más datos para entrenar, lo cual hace más robusto su entrenamiento y proporciona mejores resultados.

Los resultados demuestran que se puede entrenar un modelo empleando CNN para clasificar arañas y predecir si estas son de importancia médica o no.



Bibliografía

Feurer M., Klein A., Eggensperger K., Springenberg J.T., Blum M., Hutter F. (2019). Auto-sklearn:Efficient and Robust Automated Machine Learning. In: Hutter F., Kotthoff L., Vanschoren J. (eds) Automated Machine Learning. The Springer Series on Challenges in Machine Learning. Springer, Cham.

