

Inteligencia Artificial

Diagnóstico de neumonía usando imágenes de rayos X de tórax, Aplicando redes neuronales convolucionales.

ARTIFICIAL INTELLIGENCE
Diagnosis of pneumonia using chest X-ray images,
Applying convolutional neural networks.

Wilder Choque Cruz
gowilder@hotmail.com

Universidad Mayor De San Andrés, Facultad de Ciencias Puras y N., Carrera de Informática, Inteligencia Artificial

Resumen

La neumonía es una enfermedad que ha causado gran cantidad de muertes a nivel mundial. Los métodos de detección de neumonía: pruebas de sangre, tomografía computarizada y las radiografías (RX) de tórax en general requieren de un médico especialista y tiempo para su análisis.

En la actualidad el mundo se encuentra enfrentado a una crisis sanitaria global provocada por la enfermedad Covid-19. Desde su aparición en Wuhan (China) el virus ha causado alrededor de cuatro millones de muertes en el mundo provocando una alta presión hospitalaria. El virus destaca por su alta capacidad de propagación. Por ello, parece indispensable contar con mecanismos de detección precoz de la enfermedad para aumentar la rapidez del tratamiento y reducir la probabilidad de contagio. Las radiografías de tórax son una prueba muy efectiva para la detección de neumonía a causa de Covid-19. El aprendizaje profundo es capaz de extraer características relacionadas con los resultados clínicos de las radiografías. En este proyecto de inteligencia artificial se ha propuesto un modelo de red neuronal convolucional y se ha intentado mejorar mediante la utilización de las técnicas de aumento de datos, transferencia de aprendizaje y ajuste fino. El proyecto se ha desarrollado en el lenguaje de programación Python mediante la utilización de las librerías Tensorflow y Keras. Para el entrenamiento del modelo se ha utilizado una base de datos de imágenes de radiografías de tórax del repositorio Kaggle (chest_xray). Dichas imágenes se encuentran clasificadas en 3 categorías: normal, neumonía bacteriana y neumonía viral. Finalmente se ha realizado una comparación de los modelos creados. El considerado mejor modelo ha sido utilizado para mostrar los resultados.

Palabras Clave: Inteligencia Artificial, redes neuronales, Diagnóstico de neumonía.

Abstract

Pneumonia is a disease that has caused a large number of deaths worldwide. Pneumonia detection methods: blood tests, computed tomography and chest X-rays (RX) generally require a specialist doctor and time for analysis.

The world is currently facing a global health crisis caused by the Covid-19 disease. Since its appearance in Wuhan (China), the virus has caused around four million deaths worldwide, causing high hospital pressure. The virus stands out for its high propagation capacity. Therefore, it seems essential to have mechanisms for early detection of the disease to increase the speed of treatment and reduce the probability of contagion. Chest x-rays are a very effective test for the detection of pneumonia due to Covid-19. Deep learning is capable of extracting features related to clinical outcomes from radiographs. In this artificial intelligence project, a convolutional neural network model has been proposed and improved by using data augmentation, transfer learning and fine tuning techniques. The project has been developed in the Python programming language using the Tensorflow and Keras libraries. For model training, a database of chest X-ray images from the Kaggle repository (chest_xray) was used. These images are classified into 3 categories: normal, bacterial pneumonia and viral pneumonia. Finally, a comparison of the created models has been made. The best model considered has been used to show the results.

Keywords: Artificial Intelligence, neural networks, Diagnosis of pneumonia.

1. INTRODUCCIÓN

La neumonía es un tipo de infección respiratoria aguda (IRA) que afecta los pulmones. Estos están formados por pequeños sacos llamados alvéolos; que en pacientes sanos se llenan de aire al respirar y en personas enfermas con neumonía están llenos de pus y líquido, lo cual genera dolor al respirar y limita el oxígeno que entra a los pulmones. La neumonía puede ser causada por diversos agentes infecciosos, como virus, bacterias y hongos. En cuanto al diagnóstico de la neumonía, las radiografías de tórax son un importante método en todo el mundo, ya que permiten observar la presencia de líquido en los pulmones, la acumulación de aire en el espacio alrededor de ellos o si existen opacidades pulmonares asociadas. Sin embargo, se requiere un médico especialista para interpretar las imágenes de rayos X de manera correcta. Las limitaciones están asociadas a incertidumbres en el diagnóstico, ya que otras enfermedades también pueden mostrar opacidades similares en las imágenes. Por este motivo, se requiere precisión en la lectura de las imágenes.

la inteligencia artificial (IA) se ha venido integrando con la medicina y se han desarrollado nuevos algoritmos que han permitido dar apoyo al diagnóstico y tratamiento de enfermedades, de esta manera se han desarrollado modelos de identificación para encontrar evidencias de neumonía en imágenes clínicas.

En este trabajo se desarrolla un algoritmo que permite el procesamiento y clasificación de imágenes médicas de radiografías de tórax con el propósito de detectar neumonía causada por virus y diferenciarla de otras afectaciones pulmonares. Para ello se hace uso del aprendizaje profundo mediante la construcción de modelos de redes neuronales convolucionales.

Para mostrar los resultados se diseña y desarrolla un programa capaz de realizar predicciones sobre una imagen de radiografía de tórax subida por el usuario.

2. TEORÍA

Dataset

El conjunto de datos está organizado en 3 carpetas (entrenamiento, prueba, val) y contiene subcarpetas para cada categoría de imagen (neumonía/normal). Hay 5.863 imágenes de rayos X (JPEG) y 2 categorías (neumonía/normal).

Se seleccionaron imágenes de rayos X de tórax (anterior-posterior) de cohortes retrospectivas de pacientes pediátricos de uno a cinco años del Centro Médico de Mujeres y Niños de Guangzhou, Guangzhou. Todas las imágenes de rayos X de tórax se realizaron como parte de la atención clínica de rutina de los pacientes.

Para el análisis de las imágenes de rayos X de tórax, todas las radiografías de tórax se revisaron inicialmente para el control de calidad mediante la eliminación de todos los escaneos de baja calidad o ilegibles. Luego, dos médicos expertos calificaron los diagnósticos de las imágenes antes de autorizarlos para entrenar el sistema de IA. Para tener en cuenta cualquier error de calificación, un tercer experto también verificó el conjunto de evaluación.

3. MATERIALES Y MÉTODOS/METODOLOGÍA

Procesamiento de datos

Primero partiendo de ¿Cómo se diagnostica?

En la neumonía, los alvéolos, que deberían estar llenos de aire, se llenan de líquido o tejido inflamatorio, por lo que, en una radiografía, esto se observa de color blanco, mientras que el espacio lleno de aire aparece de color oscuro. La presencia de este color blanco confirma el diagnóstico de la infección.



Enfermo

Sano

Por consiguiente, teniendo claro estos conceptos fundamentales de Inteligencia artificial: ML y DL

- **IA (Inteligencia Artificial):** una máquina que es capaz de imitar el razonamiento humano.
- **ML (Machine Learning):** un subconjunto de Inteligencia Artificial donde las personas «entrenan» a las máquinas para reconocer patrones basados en datos y hacer sus predicciones.
- **DL (Deep Learning):** un subconjunto de ML en el que la máquina es capaz de razonar y sacar sus propias conclusiones, aprendiendo por sí misma. Sacar sus propias conclusiones, aprendiendo por sí misma.

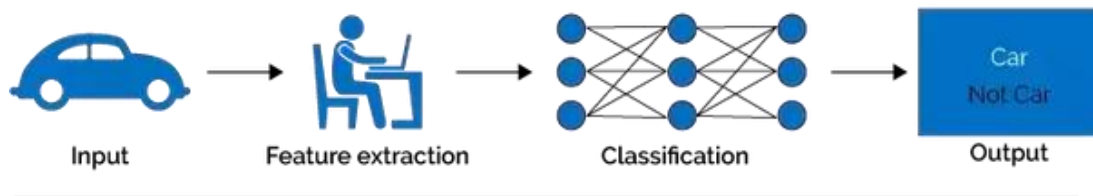
La IA (Inteligencia Artificial)



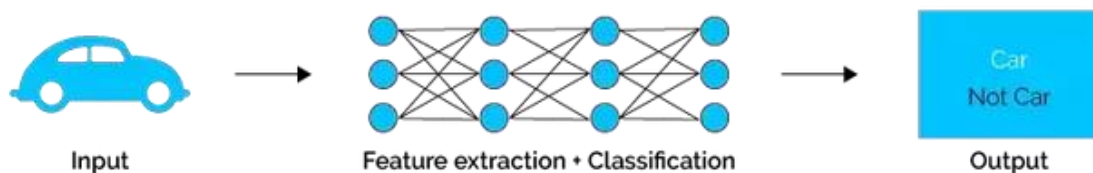
ML (Machine Learning)



Machine Learning

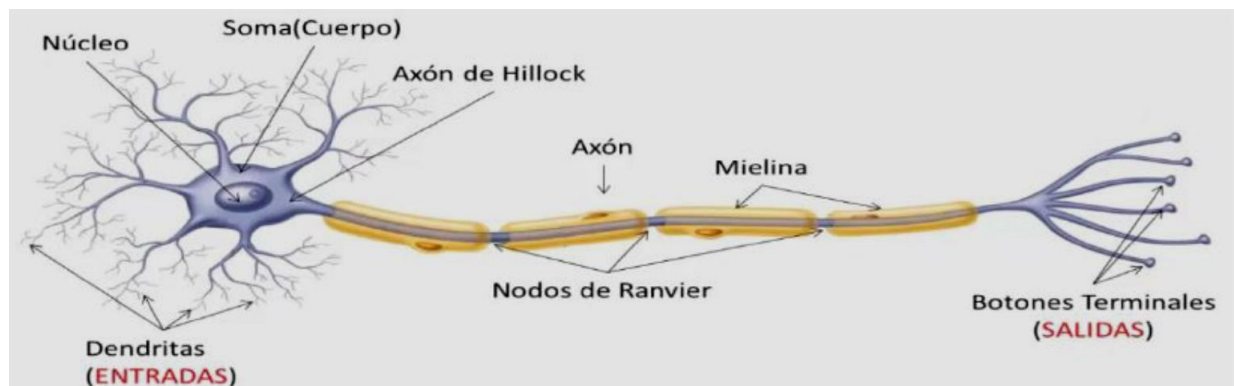


Deep Learning



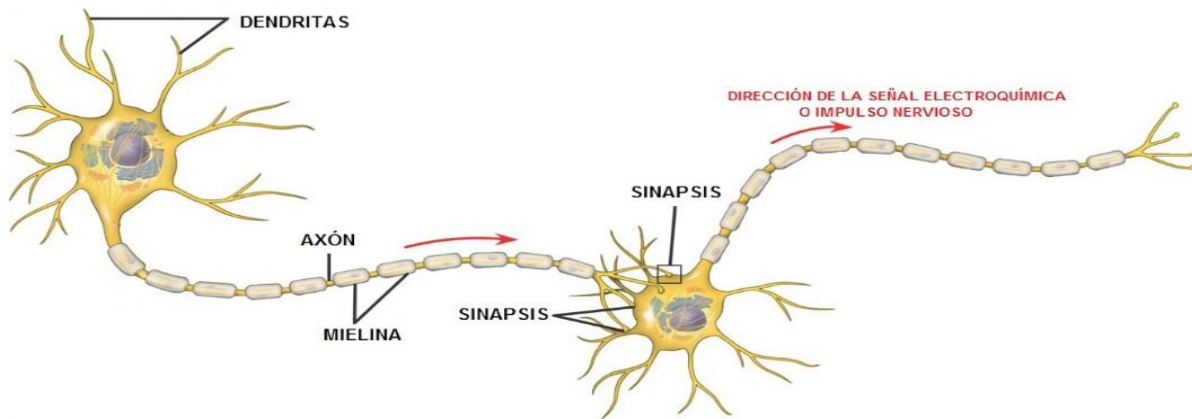
Redes neuronales artificiales

Una red neuronal artificial de un algoritmo que funciona como los científicos consideran que funciona nuestro cerebro. Nuestro sistema nervioso está constituido por células llamadas neuronas, que se comunican unas con otras a través de caminos llamados sinapsis. Estas células forman extensas redes que procesan la información de nuestro cuerpo para la toma de decisiones.



Sinapsis

La sinapsis es la manera que se comunican y organizan las neuronas y las divisiones del sistema nervioso. Las sinapsis ocurren en la corteza del cerebro donde se encuentran las células nerviosas o también llamadas neuronas. La conexión funcional entre una neurona y una segunda célula se llama sinapsis.



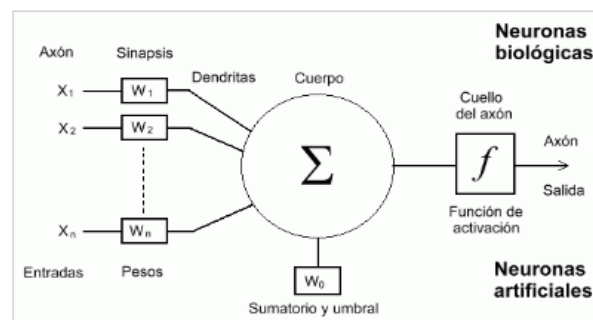
Las redes neuronales artificiales (también conocidas como sistemas conexionistas) se trata de modelo computacional evolucionado a partir de diversas aportaciones científicas que están registradas en la historia.¹ Consiste en un conjunto de unidades, llamadas neuronas artificiales, conectadas entre sí para transmitirse señales. La información de entrada atraviesa la red neuronal (donde se somete a diversas operaciones) produciendo unos valores de salida.

Modelo neuronal de McCulloch-Pitts

El primer modelo matemático de una neurona artificial, creado con el fin de llevar a cabo tareas simples, fue presentado en el año 1943 en un trabajo conjunto entre el psiquiatra y neuroanatomista Warren McCulloch y el matemático Walter Pitts.

La siguiente figura muestra un ejemplo de modelo neuronal con n entradas, que consta de:

- Un conjunto de entradas x_1, \dots, x_n .
- Los pesos sinápticos w_1, \dots, w_n , correspondientes a cada entrada.
- Una función de agregación, Σ .
- Una función de activación, f .
- Una salida, Y .



Las entradas son el estímulo que la neurona artificial recibe del entorno que la rodea, y la salida es la respuesta a tal estímulo. La neurona puede adaptarse al medio circundante y aprender de él modificando el valor de sus pesos sinápticos, y por ello son conocidos como los parámetros libres del modelo, ya que pueden ser modificados y adaptados para realizar una tarea determinada.

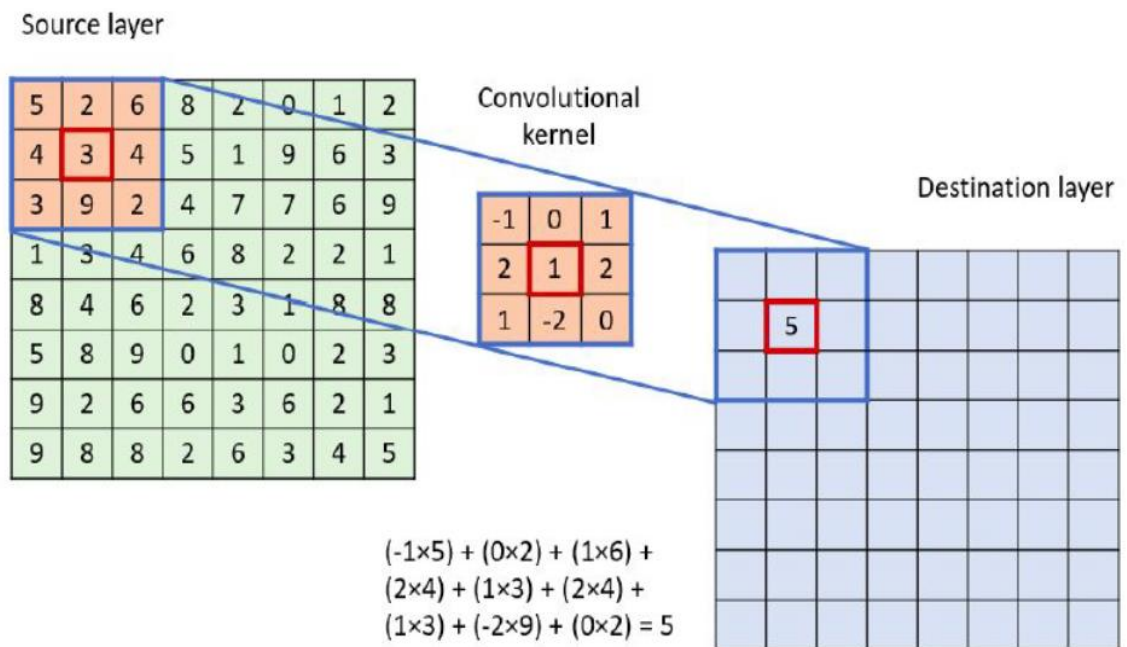
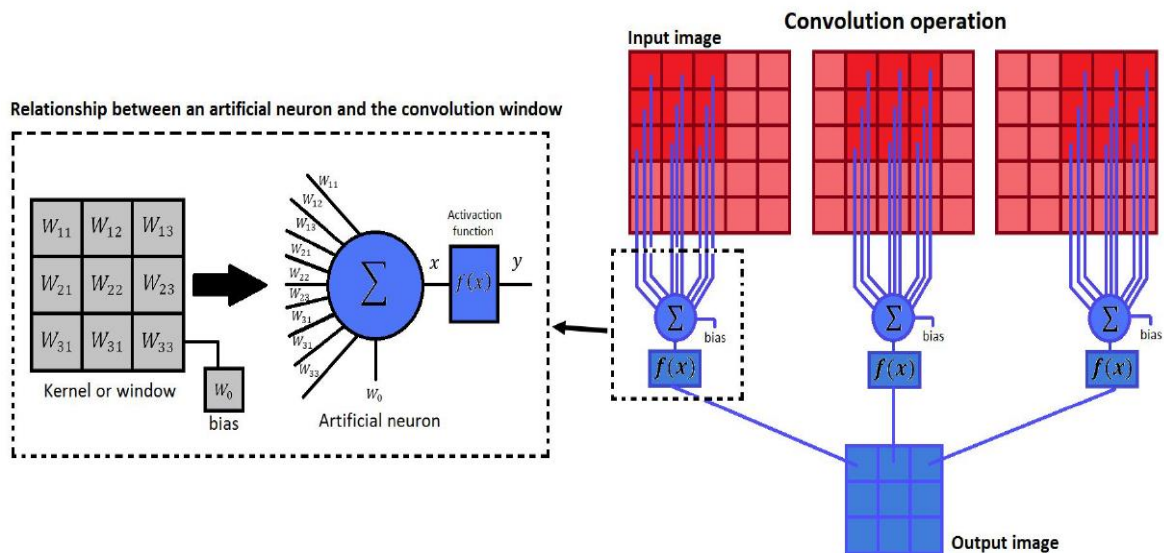
En este modelo, la salida neuronal Y está dada por:

$$Y = f\left(\sum_{i=1}^n w_i x_i\right)$$

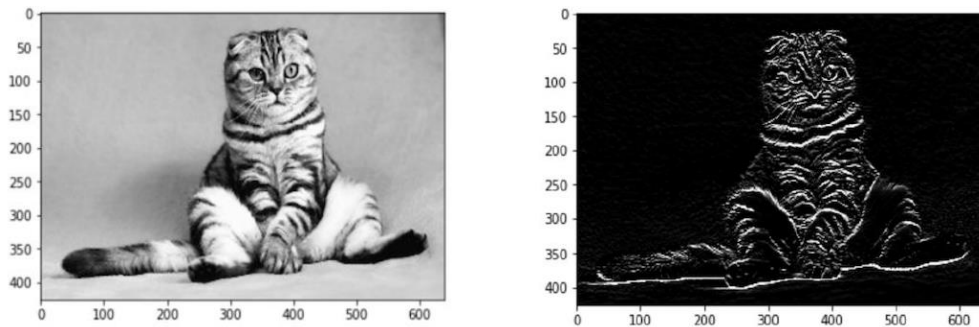
Convolución

En teoría de señales, la convolución es una de las operaciones más importantes. En particular en el procesamiento digital de imágenes, la operación de convolución permite realizar un procedimiento llamado filtrado. Dicho procedimiento es llevado a cabo “haciendo pasar” una ventana de convolución (kernel) sobre toda la imagen y realizando una suma ponderada de sus elementos.

Red Neuronal Convencional (Convolutional Neural Network)

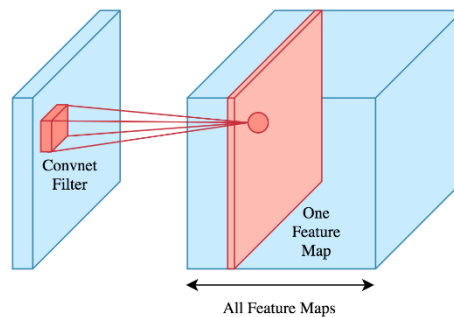


Estos filtros lo que hacen es resaltar características de los datos. En el caso de imágenes, esas características son comúnmente bordes, esquinas, cambio de nivel de color, entre otras. Cada capa convolucional genera nuevas imágenes llamadas mapas de rasgos.

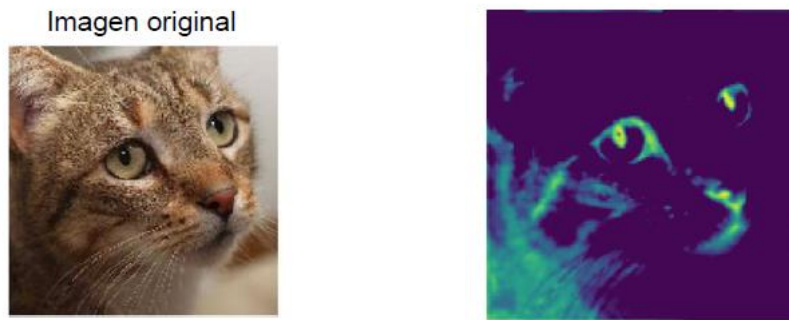


Visualización de mapas de características

Recapitemos rápidamente la arquitectura de convolución como recordatorio. Un filtro convnet opera en la entrada que realiza la operación de convolución y, como resultado, obtenemos un mapa de entidades. Utilizamos múltiples filtros y apilamos los mapas de entidades resultantes para obtener un volumen de salida. Primero visualizaremos los mapas de entidades, y en la siguiente sección exploraremos los filtros convnet.

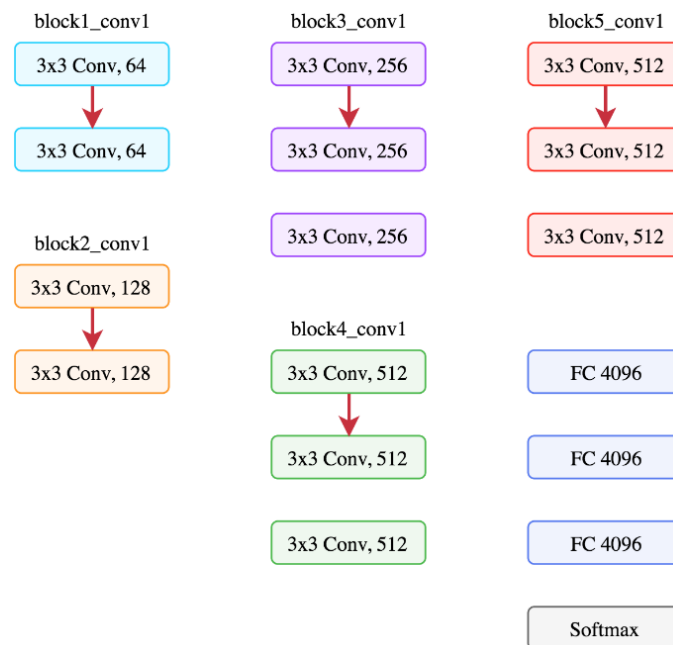


Visualizaremos los mapas de entidades para ver cómo se transforma la entrada pasando por las capas de convolución. Los mapas de entidades también se denominan activaciones intermedias, ya que la salida de una capa se denomina activación. Recuerde que la salida de una capa de convolución es un volumen 3D. Como discutimos anteriormente, la altura y el ancho corresponden a las dimensiones del mapa de entidades, y cada canal de profundidad es un mapa de entidades distinto que codifica entidades independientes. Por lo tanto, visualizaremos mapas de entidades individuales trazando cada canal como una imagen 2D. Cómo visualizar los mapas de características es en realidad bastante simple. Pasamos una imagen de entrada a través de la CNN y grabamos las activaciones intermedias. Luego seleccionamos aleatoriamente algunos de los mapas de entidades y los trazamos. Las capas convolucionales VGG se denominan de la siguiente manera: blockX_convY. Por ejemplo, el segundo filtro en el tercer bloque de convolución se llama block3_conv2. En el diagrama de arquitectura anterior corresponde al segundo filtro púrpura. Por ejemplo, uno de los mapas de entidades de la salida de la primera capa (block1_conv1) tiene el siguiente aspecto.

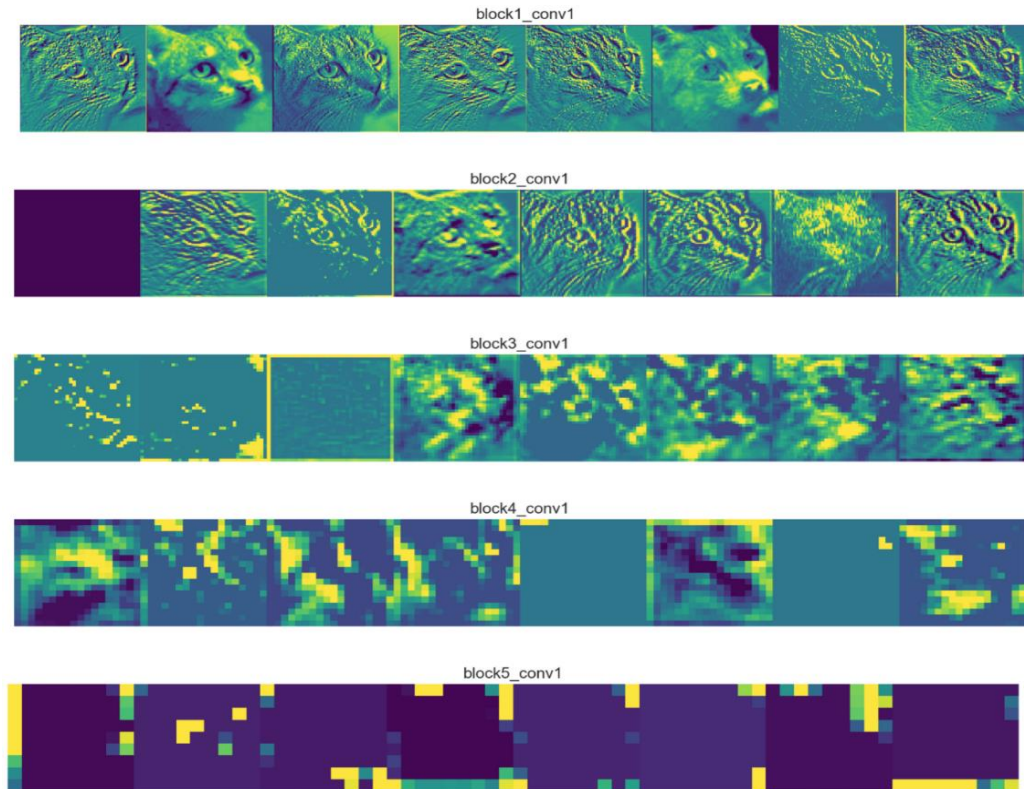


Las áreas brillantes son las regiones "activadas", lo que significa que el filtro detectó el patrón que estaba buscando. Este filtro parece codificar un detector de ojos y nariz.

En lugar de mirar un solo mapa de entidades, sería más interesante visualizar múltiples mapas de entidades desde una capa de convolución. Así que visualicemos los mapas de entidades correspondientes a la primera convolución de cada bloque, las flechas rojas en la figura a continuación.

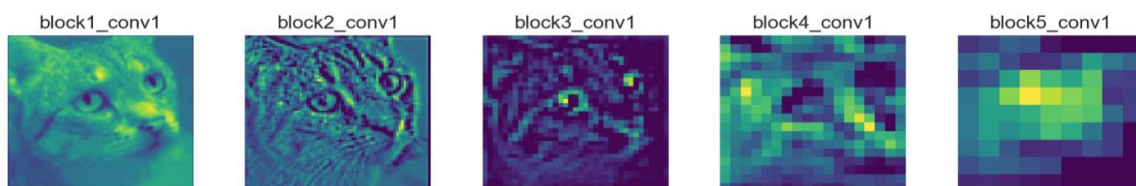


La siguiente figura muestra 8 mapas de entidades por capa. Block1_conv1 en realidad contiene 64 mapas de entidades, ya que tenemos 64 filtros en esa capa. Pero solo estamos visualizando los primeros 8 por capa en esta figura.



Hay algunas observaciones interesantes sobre los mapas de entidades a medida que avanzamos a través de las capas. Echemos un vistazo a un mapa de entidades por capa para hacerlo más obvio.

Los mapas de entidades de la primera capa (block1_conv1) conservan la mayor parte de la información presente en la imagen. En las arquitecturas CNN las primeras capas suelen actuar como detectores de bordes.



A medida que profundizamos en la red, los mapas de características se parecen menos a la imagen original y más a una representación abstracta de la misma. Como puedes ver en block3_conv1 el gato es algo visible, pero después de eso se vuelve irreconocible. La razón es que los mapas de características más profundos codifican conceptos de alto nivel como "nariz de gato" u "oreja de perro", mientras que los mapas de características de nivel inferior detectan bordes y formas simples. Es por eso que los mapas de entidades más profundos contienen menos información sobre la imagen y más sobre la clase de la imagen. Todavía codifican características útiles, pero son menos interpretables visualmente por nosotros.

Los mapas de entidades se vuelven más escasos a medida que profundizamos, lo que significa que los filtros detectan menos funciones. Tiene sentido porque los filtros en las primeras capas detectan formas simples, y cada imagen las contiene. Pero a medida que profundizamos, comenzamos a buscar cosas más complejas como "cola de perro" y no aparecen en todas las imágenes. Es por eso que en la primera figura con 8 filtros por capa, vemos más de los mapas de entidades en blanco a medida que profundizamos (block4_conv1 y block5_conv1).

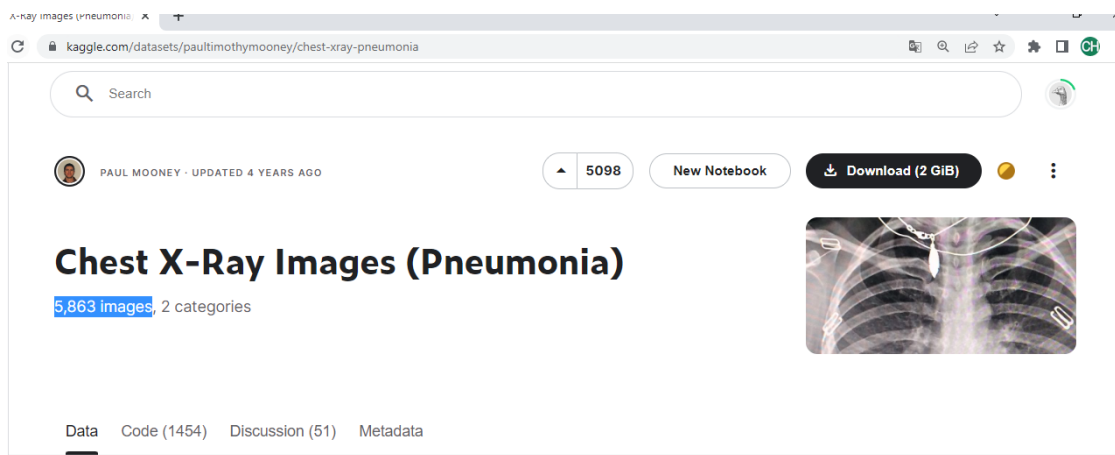
¿Qué es el aprendizaje de una red neuronal?

Los seres humanos cuando son bebés, no aprenden a caminar de inmediato. El aprendizaje se realiza durante un cierto periodo de tiempo y después de un proceso de repetición. Se comienza con asimilar pequeños componentes del movimiento del cuerpo humano y se ajusta con retroalimentación del ambiente, hasta que por fin se logra caminar lo suficientemente bien.



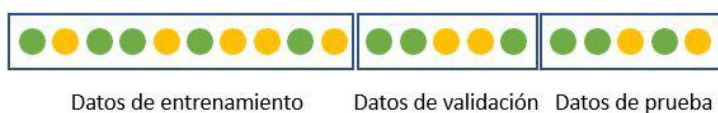
2. RESULTADOS Y DISCUSIÓN

En esta investigación se evalúa con una cantidad de más de 5000 imágenes de rayos X de pacientes de dos clases distintas: Neumonía/Normal



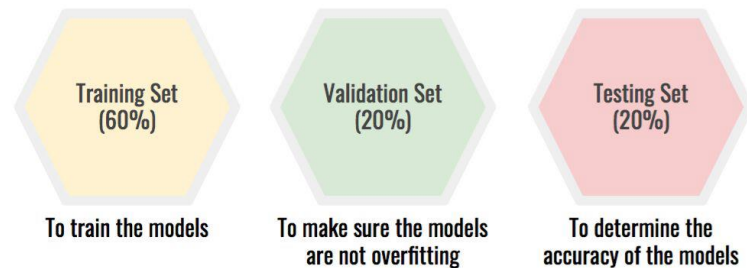
<https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia>

El dataset original contiene muy pocas imágenes en el conjunto de validación, por lo que se modificó para contar con más datos para validación y así evitar el sobreajuste.



Se dividió de manera porcentual.

- Entrenamiento 60%
- Validación 20%
- Prueba 20%



LA RED NEURONAL CONVOLUCIONAL

Esta red está compuesta por tres capas convolucionales, tres de pooling, una de flatten y dos capas densas. Muestra los detalles de la arquitectura propuesta.

Resultado esperado:

```
[ ] # Detalle de la red neuronal convolucional
    cnn.summary()
```

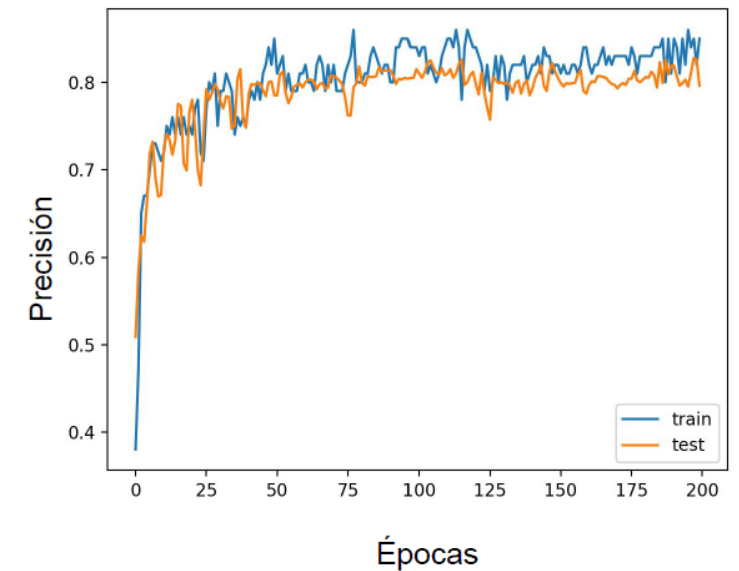
Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d (MaxPooling2D)	(None, 74, 74, 32)	0
conv2d_1 (Conv2D)	(None, 72, 72, 32)	9248
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 32)	0
conv2d_2 (Conv2D)	(None, 34, 34, 32)	9248
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 32)	0
flatten (Flatten)	(None, 9248)	0
dense (Dense)	(None, 128)	1183872
dense_1 (Dense)	(None, 1)	129

```

Total params: 1,203,393
Trainable params: 1,203,393
Non-trainable params: 0
    
```

De una manera similar, las redes neuronales se entrenan realizando repeticiones llamadas iteraciones. Cuando una red procesa todos los datos del conjunto de entrenamiento se le conoce como época. Es importante hacer notar que no hay manera de pre-calculer cuántas épocas se necesitan y este es un hiperparámetro del modelo que debe ser ajustado para cada problema.



PREPROCESAMIENTO DE LAS IMÁGENES

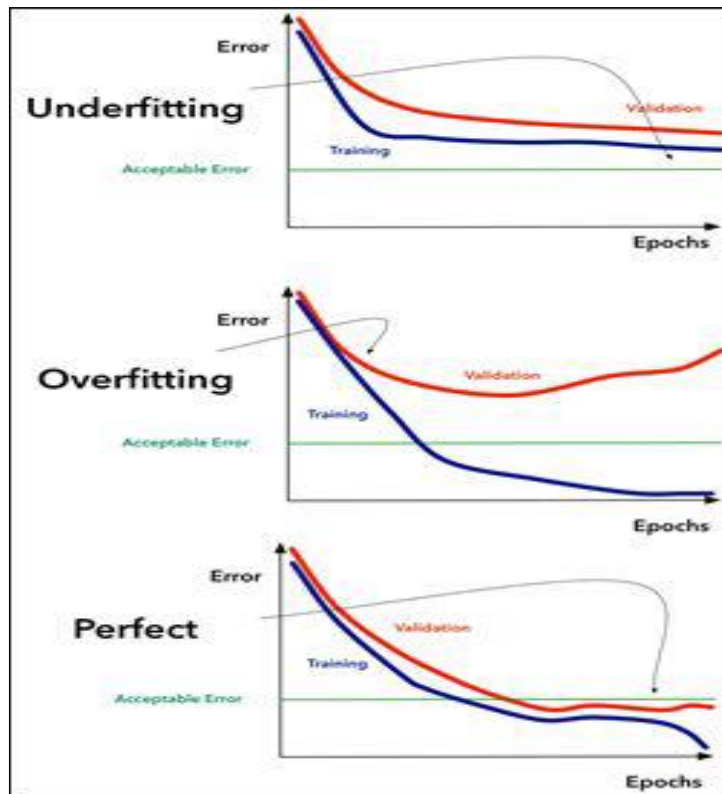
Utilice lotes de 20 imágenes y considere que se trata de un modelo binario. Entrene la CNN utilizando 10 épocas, 100 pasos por época y 50 pasos de validación.

Resultado esperado:

```
cnn_model = cnn.fit(training_set,
                    steps_per_epoch=100,
                    epochs=10,
                    validation_data=validation_generator,
                    validation_steps=50)
```

Epoch 1/10
100/100 [=====] - 49s 372ms/step - loss: 0.4523 - accuracy: 0.8015 - val_loss: 0.2001 - val_accuracy: 0.9420
Epoch 2/10
100/100 [=====] - 37s 372ms/step - loss: 0.3119 - accuracy: 0.8735 - val_loss: 0.1843 - val_accuracy: 0.9290
Epoch 3/10
100/100 [=====] - 37s 370ms/step - loss: 0.2522 - accuracy: 0.9046 - val_loss: 0.1685 - val_accuracy: 0.9410
Epoch 4/10
100/100 [=====] - 39s 388ms/step - loss: 0.2544 - accuracy: 0.9011 - val_loss: 0.1815 - val_accuracy: 0.9250
Epoch 5/10
100/100 [=====] - 36s 365ms/step - loss: 0.2212 - accuracy: 0.9152 - val_loss: 0.1230 - val_accuracy: 0.9610
Epoch 6/10
100/100 [=====] - 38s 383ms/step - loss: 0.2072 - accuracy: 0.9210 - val_loss: 0.1006 - val_accuracy: 0.9680
Epoch 7/10
100/100 [=====] - 37s 369ms/step - loss: 0.1888 - accuracy: 0.9280 - val_loss: 0.1216 - val_accuracy: 0.9570
Epoch 8/10
100/100 [=====] - 37s 371ms/step - loss: 0.1832 - accuracy: 0.9262 - val_loss: 0.1841 - val_accuracy: 0.9330
Epoch 9/10
100/100 [=====] - 37s 369ms/step - loss: 0.1921 - accuracy: 0.9267 - val_loss: 0.1389 - val_accuracy: 0.9570
Epoch 10/10
100/100 [=====] - 38s 381ms/step - loss: 0.1673 - accuracy: 0.9370 - val_loss: 0.1213 - val_accuracy: 0.9530

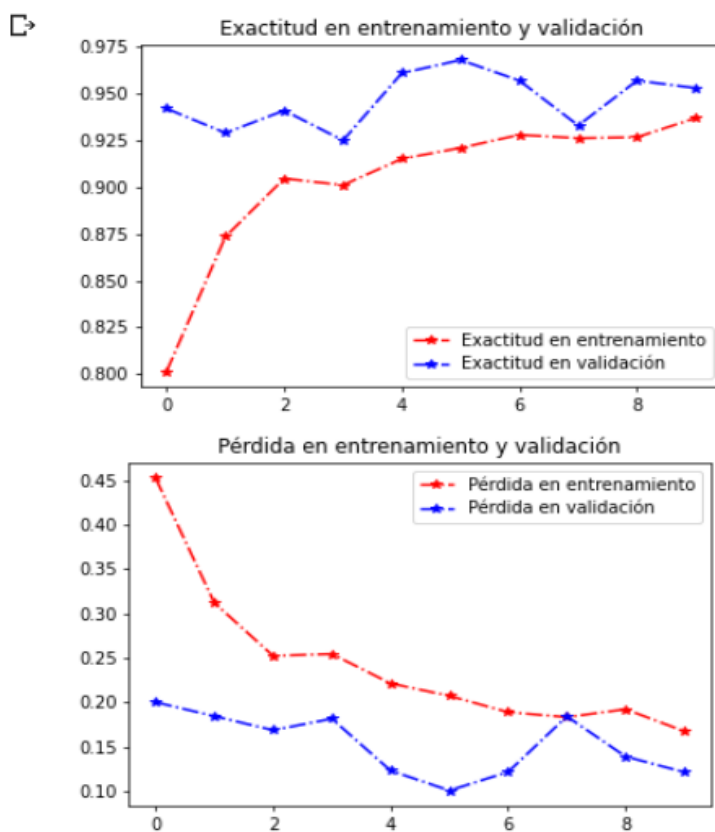
Durante el proceso de entrenamiento, se pueden presentar dos circunstancias relacionadas con el desempeño. En primer lugar, se tiene el underfitting en la cual de función de pérdida de entrenamiento y validación se encuentran por encima de un umbral aceptable. En otras palabras, el modelo ya no está aprendiendo. En el caso del overfitting la curva de validación y entrenamiento se empiezan a separar, esto indica que el modelo no está generalizando. Lo ideal es que la curva de validación y entrenamiento se encuentren por debajo de un umbral y no se separen.



Se grafico la exactitud y la pérdida en el entrenamiento y validación.

Resultado esperado:

```
plt.show()
```



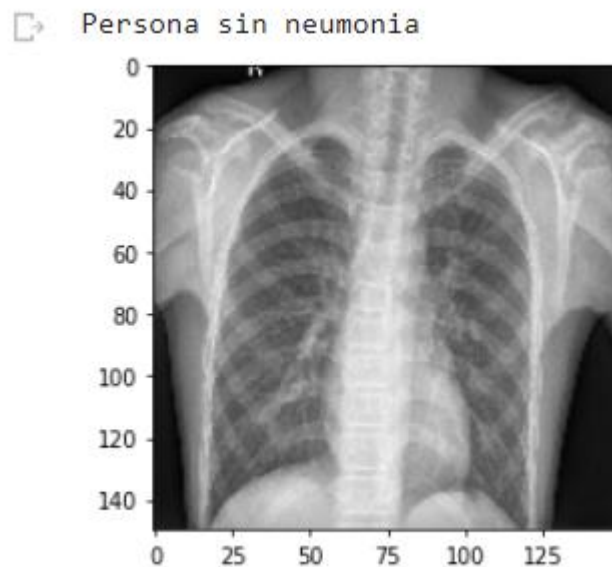
Se evaluó el desempeño del modelo generado con el conjunto de prueba. Utilizando el modelo generado para predecir una imagen, si la persona tiene o no neumonía.

```
test_accu = cnn.evaluate(test_set, steps=50)

50/50 [=====] - 8s 158ms/step - loss: 0.2914 - accuracy: 0.9040

[ ] print('La exactitud en el conjunto de prueba es: ', test_accu[1]*100, '%')

La exactitud en el conjunto de prueba es: 90.39999842643738 %
```



3. CONCLUSIONES

- El deep learning es una de las áreas de mayor crecimiento en los últimos años y ha provocado una revolución en la comunidad de inteligencia artificial.
- Detrás del aprendizaje profundo se encuentran las redes neuronales artificiales, que son algoritmos que buscan emular el comportamiento de las neuronas reales.
- Bibliotecas como TensorFlow y Keras permiten a los desarrolladores implementar algoritmos de deep learning en unas cuantas líneas de código.
- Las redes neuronales convolucionales son el modelo de deep learning más utilizado y son ampliamente utilizadas en tareas de visión por computadora.

En esta investigación se presenta una comparación exhaustiva entre las principales arquitecturas de CNN como clasificadores de neumonía, con el objetivo de determinar que arquitecturas son las más confiables para este diagnóstico, y cuáles suelen trabajar de mejor manera (aprender características) con imágenes radiológicas de pacientes con neumonía.

Se pudo hallar una exactitud en el conjunto de prueba de 90.399%, lo que garantiza que las métricas de rendimiento tuvieron un excelente equilibrio entre las predicciones.

Se espera que esta investigación pueda servir como guía para la comunidad científica y a futuros desarrolladores de sistemas inteligente para el apoyo a médicos expertos en el diagnóstico de neumonía.

4. REFERENCIAS

1. **Petrosillo, N., Viceconte, G., Ergonul, O., Ippolito, G., Petersen, E. (2020).** COVID-19, SARS and MERS: are they closely related? *Clinical Microbiology and Infection*. DOI: 10.1016/j.cmi.2020.03.026. [[Links](#)]
2. **Lippi, G., Plebani, M., Henry, B.M. (2020).** Thrombocytopenia is associated with severe coronavirus disease 2019 (COVID-19) infections: A meta-analysis. *Clinica Chimica Acta*, Vol. 506, pp. 145–148. DOI: 10.1016/j.cca.2020.03.022. [[Links](#)]
3. **Zhou, Z., Guo, D., Li, C., Fang, Z., Chen, L., Yang, R., Zeng, W. (2020).** Coronavirus disease 2019: initial chest CT findings. *European Radiology*. DOI: 10.1007/s00330-020-06816-7. [[Links](#)]
4. **Ma, Y., Zhao, Y., Liu, J., He, X., Wang, B., Fu, S., Luo, B. (2020).** Effects of temperature variation and humidity on the death of COVID-19 in Wuhan, China. *Science of The Total Environment*. DOI: 10.1016/j.scitotenv.2020.138226. [[Links](#)]
5. **Bishop, C. M.** (2006). *Pattern recognition and machine learning*. springer.
6. **Joshi.** (2017). *Artificial intelligence with python*. Packt Publishing Ltd.