

Algoritmos de *deep learning* para la detección de neumonía en infantes a través de imágenes de radiografías del tórax

Juan Carlos Valero Gómez
jvalerog@unam.edu.pe

Alex Peter Zúñiga Incalla
azunigai@unam.edu.pe

Juan Carlos Clares Perca
jclaresp@unam.edu.pe

Universidad Nacional de Moquegua, Ilo, Perú

Recibido: 20/6/2021 Aceptado: 21/9/2021
doi: <https://doi.org/10.26439/ciis2021.5586>

RESUMEN. Una gran cantidad de infantes fallecen cada año a consecuencia de la neumonía en todo el mundo. Se reporta que aproximadamente más de un millón de casos de neumonía, dentro de este grupo, se da entre 0 y 5 años de edad, de los cuales 808 694 murieron en el 2017. Por ende, la neumonía es una de las principales causas de fallecimiento entre los niños, con un alto nivel de mortalidad en Asia y África. Incluso en un país desarrollado como Estados Unidos, esta enfermedad se encuentra entre las diez principales causas de muerte. La detección y el tratamiento tempranos de la neumonía pueden reducir significativamente las tasas de mortalidad entre los infantes en países emergentes. Por lo tanto, este trabajo presenta algoritmos de *deep learning* para detectar neumonía mediante imágenes de radiografías. Se entrenaron tres algoritmos de *deep learning* para clasificar las imágenes de radiografías en dos tipos: neumonía y normal. Se presentan tres algoritmos, a cada uno se añadió una capa *pooling* de 4×4 , se vectorizaron los datos con la técnica *flatten*, se agregaron seis capas *dense* de 1024, 512, 256, 128, 64 y 32 de valor de salida, y cada una con activación *relu*; asimismo, se aplicó un BatchNormalization y, finalmente, se agregó una capa *dense* de 2 con una activación *softmax* para la clasificación. Los tres algoritmos son modelos previamente entrenados, que son Xception, MobileNet e InceptionV3 y que obtuvieron en la métrica de *accuracy* 94,4 %, 96,2 % y 95,3 % respectivamente.

PALABRAS CLAVE: neumonía infantil / *deep learning* / Xceptio / MobileNet / InceptionV3

DEEP LEARNING ALGORITHMS FOR THE DETECTION OF PNEUMONIA IN INFANTS THROUGH CHEST X-RAY IMAGES

ABSTRACT. Worldwide, large numbers of infants die of pneumonia each year. It is reported that approximately more than 1 million cases of pneumonia in infants occur between 0 and 5 years of age, of which 920 136 died in 2015. Therefore, pneumonia is one of the leading causes of death among infants, with a high level of mortality in Asia and Africa. Even in a developed country like the United States, pneumonia is among the top 10 causes of death. Early detection and treatment of pneumonia can significantly reduce mortality rates among infants in emerging countries. Therefore, this work presents deep learning algorithms to detect pneumonia using radiographic images. Three deep learning algorithms were trained to classify X-ray images into two classes: pneumonia and normal. Three algorithms are presented, to each one, a 4x4 pooling layer was added, the data is vectorized with the flatten technique, six dense layers of 1024, 512, 256, 128, 64, and 32 of output value were added, and each one with relu activation; A BatchNormalization is applied; finally a dense layer of 2 is added with a softmax activation for classification. The three algorithms are previously trained models, which are Xception, MobileNet and InceptionV3 obtained in the accuracy metric 94,4%, 96,2% and 95,3% respectively.

KEYWORDS: childhood pneumonia / deep learning / Xception / MobileNet / InceptionV3

1. INTRODUCCIÓN

La neumonía es una infección aguda respiratoria que ataca a los pulmones. La neumonía es la principal causa individual de mortalidad infantil en todo el mundo. Se calcula que la neumonía mató a unos 808 694 niños menores de cinco años en el 2017, lo que supone el 15 % de todos los fallecimientos de infantes menores de cinco años a nivel mundial (WHO, 2019). En los países emergentes hay una escasa disponibilidad de profesionales médicos y radiólogos que cumplen un papel fundamental en el diagnóstico de la neumonía (Liu *et al.*, 2018). En la última década se ha empleado para el diagnóstico de enfermedades la asistencia por computadora basada en algoritmos de inteligencia artificial; estas tecnologías se pueden poner a disposición de una gran población a un costo mínimo. Un problema recurrente en la detección de enfermedades pulmonares es que, a veces, los patrones característicos de una enfermedad tienen algún grado de similitud con otras enfermedades, ocasionando que los médicos especializados tengan dificultades en realizar un diagnóstico preciso.

Entre las técnicas de *deep learning* (aprendizaje profundo), las redes neuronales convolucionales han demostrado ser eficientes en la clasificación de imágenes, son frecuentemente utilizadas por los investigadores para la visión computacional. Estas técnicas han comprobado ser tan útiles para el reconocimiento rápido y preciso de la neumonía que se equiparan o superan a un especialista médico. No obstante, las técnicas de *deep learning* no pueden sustituir a los médicos especialistas para realizar un diagnóstico, sino que son una herramienta más para el personal médico en la toma de decisiones clínicas (Hosny *et al.*, 2018; Kallianos *et al.*, 2019).

El presente trabajo emplea tres modelos basados en algoritmos de *deep learning* que son capaces de detectar automáticamente si un paciente padece de neumonía o no. En ese sentido, se plantea desarrollar tres algoritmos de *deep learning* que puedan superar las expectativas de la investigación; para ello se empleará algoritmos preentrenados y se importará los pesos de ImageNet para lograr un mejor desempeño (Deng *et al.*, 2014). Los resultados serán evaluados a través de métricas que determinarán si los algoritmos propuestos tienen resultados óptimos.

2. METODOLOGÍA

2.1 *Dataset*

En el presente trabajo se empleó un *dataset* publicado en la comunidad de ciencias de datos KAGGLE por Kermany *et al.* (2018); el *dataset* cuenta con un total de 5856 imágenes de radiografías del tórax de infantes anonimizados. El *dataset* cuenta con dos clases: pacientes con neumonía con un total de 4273 imágenes y pacientes sanos con un total de 1583 imágenes. Como se puede visualizar en la figura 1, el *dataset* se encuentra desbalanceado; no obstante, se cuenta con un número aceptable en pacientes sanos para la experimentación; en los resultados se podrá determinar el nivel de incidencia del desbalance en las métricas de evaluación.

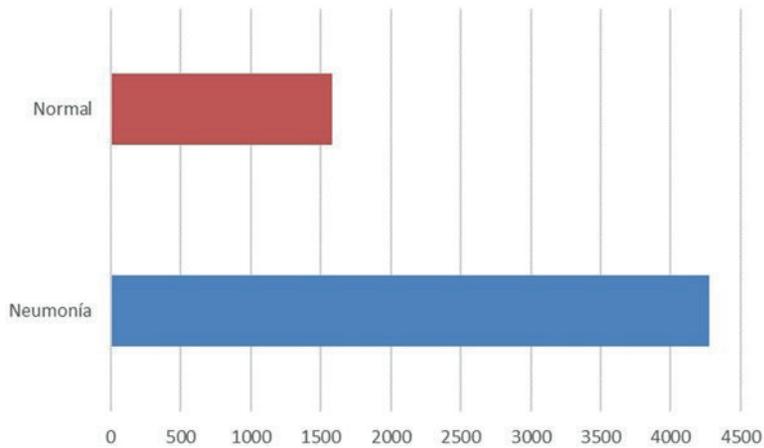


Figura 1. Distribución del dataset

Elaboración propia

El dataset se dividió en dos conjuntos: el conjunto de entrenamiento, con el 80 % del total (4696 imágenes), de las cuales 3458 imágenes son de pacientes con neumonía y 1238 imágenes de pacientes sanos; y el conjunto de test, con el 20 % del total (1160 imágenes), de las cuales 815 imágenes son de pacientes con neumonía y 345 imágenes son de pacientes sanos.

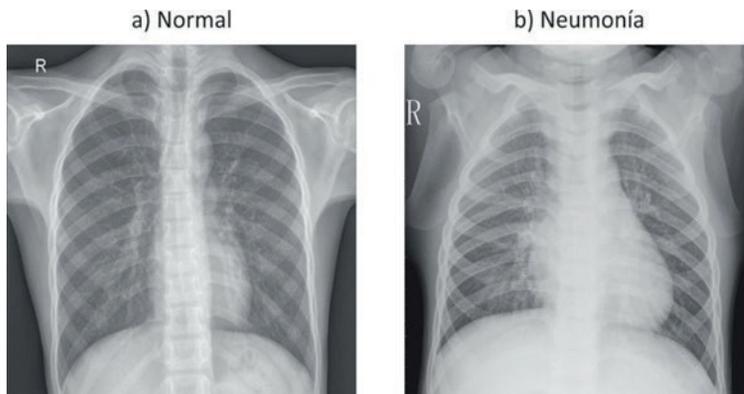


Figura 2. Distribución del dataset

Elaboración propia

Los datos del entrenamiento se dividen nuevamente en datos para entrenar y datos para la validación; en este caso del total de los datos de entrenamiento un 20 % se destinará para la validación cruzada, está conformado por 692 imágenes de pacientes con neumonía y 248 imágenes de pacientes sanos. Se puede ver una muestra en la figura 2.

Cada imagen del *dataset* ha sido preprocesada normalizándola para que pueda ser empleada en el entrenamiento de los algoritmos. Cada imagen fue reescalada a una resolución de 256 x 256 píxeles, esta resolución es aceptada por los tres algoritmos como valores de entrada, Además, cada píxel tiene un valor que se encuentra entre el rango de [0, 255], el valor del píxel ha sido convertido para que esté dentro del rango de [0, 1], esta conversión tiene como objetivo facilitar las operaciones matriciales.

2.2 Algoritmos

Se emplearon tres algoritmos de *deep learning* para el estudio: InceptionV3 (Szegedy *et al.*, 2016), MobileNet (Howard *et al.*, 2019) y Xception (Chollet, 2017). Para hacer uso de los algoritmos propuestos se utilizó la librería de Tensorflow y Keras. Para recortar los tiempos de entrenamiento de los algoritmos y una mejor eficacia de estos se utilizó la técnica de Transfer Learning con ImageNet (Deng *et al.*, 2014). El hecho de usar la misma arquitectura con los mismos parámetros que el modelo preentrenado otorgará a los algoritmos la capacidad de reconocer varias clases desde el comienzo, lo que además se traducirá en un tiempo de entrenamiento muy reducido. Para tener la capacidad de computo necesaria para ejecutar los algoritmos InceptionV3, MobileNet y Xception se empleó la plataforma de Google Colab que proporciona el uso de los GPU necesarios para la experimentación.

A la arquitectura del algoritmo Xception se sumó una capa *pooling* de 4 x 4, se vectorizaron los datos con la técnica *flatten*, se agregaron seis capas *dense* de 1024, 512, 256, 128, 64 y 32 de valor de salida y cada una con activación *relu*; se aplicó un BatchNormalization; finalmente, se agregó una capa *dense* de 2 con una activación *softmax* para la clasificación.

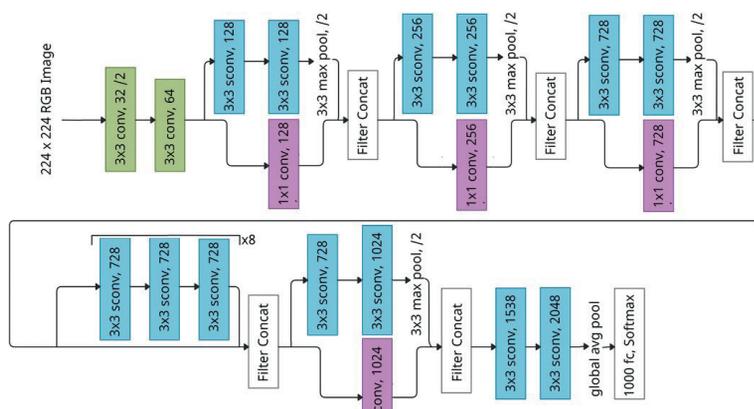


Figura 3. Arquitectura de Xception

Fuente: Adaptado del sitio <http://www.techscience.com>

A la arquitectura del algoritmo MobileNet se le sumó una capa *pooling* de 4 x 4, se vectorizaron los datos con la técnica *flatten*, se agregaron seis capas *dense* de 1024, 512, 256, 128, 64 y 32 de valor de salida y cada una con activación *relu*; se aplicó un BatchNormalization, finalmente se le agregó una capa *dense* de 2 con una activación *softmax* para la clasificación.

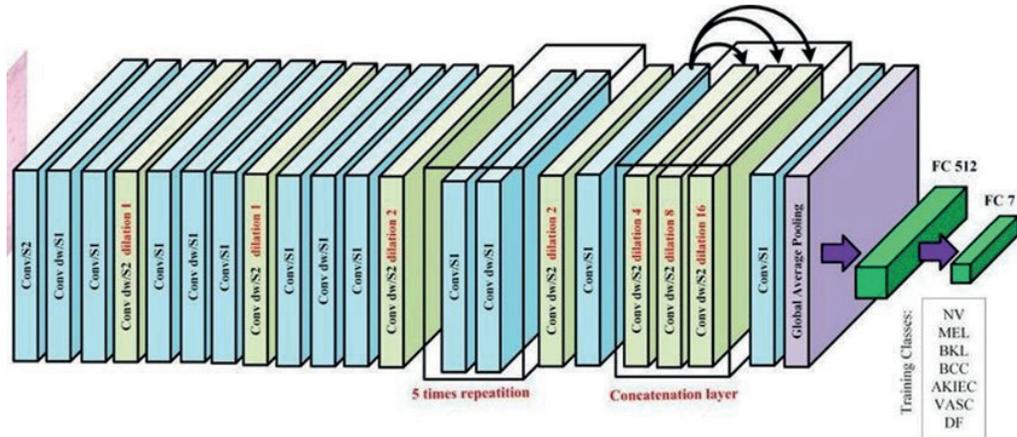


Figura 4. Arquitectura de MobileNet
Elaboración propia

A la arquitectura del algoritmo InceptionV3 se le sumó una capa *pooling* de 4 x 4, se vectorizaron los datos con la técnica *flatten*, se agregaron seis capas *dense* de 1024, 512, 256, 128, 64 y 32 de valor de salida y cada una con activación *relu*; se aplicó un BatchNormalization, finalmente se agregó una capa *dense* de 2 con una activación *softmax* para la clasificación.

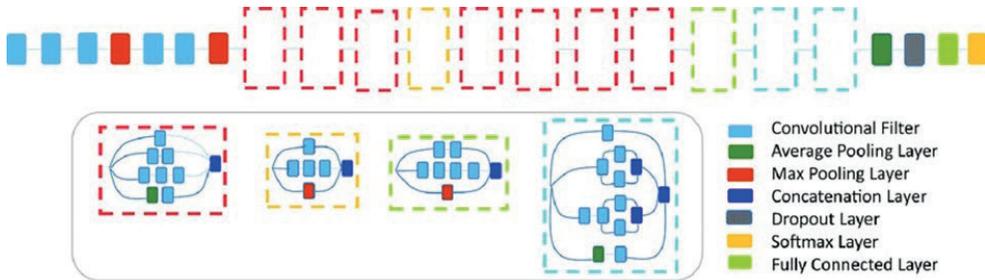


Figura 5. Arquitectura de InceptionV3
Elaboración propia

Los hiperparámetros empleados para los tres algoritmos fueron epochs = 100, batch_size = 20 y el optimizador SGD (Song *et al.*, 2013) con learning_rate a razón de 0,001. Esta configuración permitió obtener el mejor desempeño que pueden ofrecer InceptionV3, MobileNet y Xception.

La selección de las capas adicionales para cada modelo es en razón de darle más profundidad y que cada modelo pueda extraer y mapear adecuadamente las imágenes del *dataset*, debido a que los modelos ya han sido preentrenados con los pesos de ImageNet.

3. RESULTADOS

En la etapa de entrenamiento se puede observar el desempeño de cada algoritmo empleado para el presente estudio, se logró evaluar a través de las métricas de precisión (*accuracy*) y la pérdida de entropía cruzada (*cross entropy loss*). En el caso de Xception se visualiza que sufre de sobreajuste (*overfitting*) considerable después de la época 15, pero se mantiene estable (véase la figura 6). Para MobileNet se visualiza que desde las primeras épocas obtiene buenos resultados sin caer en el sobreajuste pronunciado, con pocas variaciones y manteniéndose constante (véase la figura 7). En el caso de InceptionV3 obtiene buenos resultados desde el inicio; además, los resultados de entrenamiento y validación son más semejantes en comparación de los resultados de Xception y MobileNet (véase la figura 8).

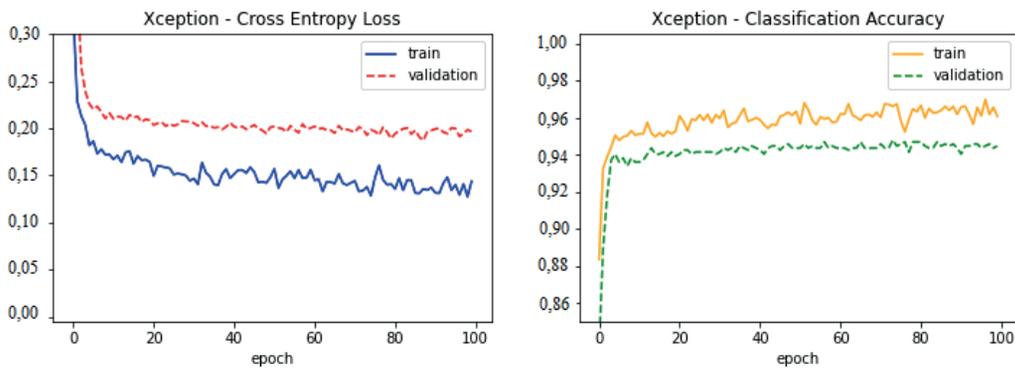


Figura 6. Evaluación del desempeño en la etapa de entrenamiento y validación de las métricas de precisión y la entropía cruzada para Xception

Elaboración propia

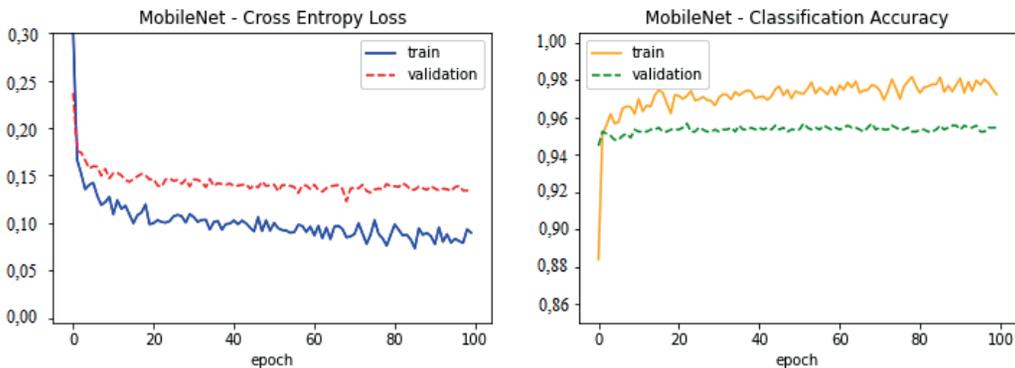


Figura 7. Evaluación del desempeño en la etapa de entrenamiento y validación de las métricas de precisión y la entropía cruzada para MobileNet

Elaboración propia

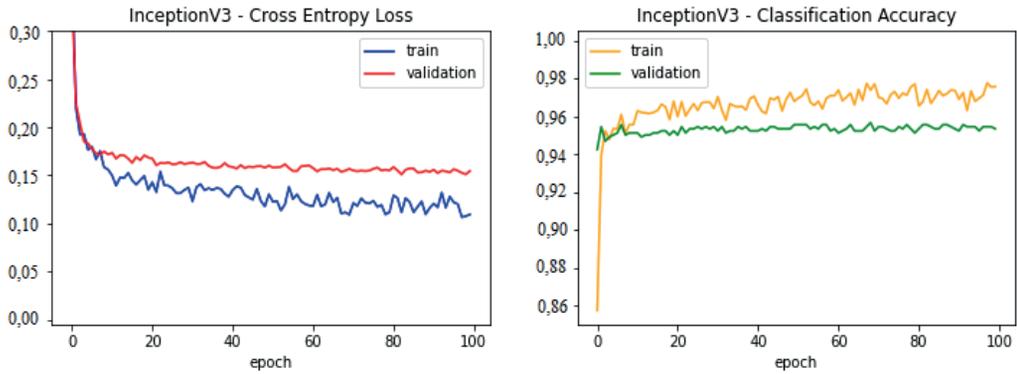


Figura 8. Evaluación del desempeño en la etapa de entrenamiento y validación de las métricas de precisión y la entropía cruzada para InceptionV3

Elaboración propia

Para la etapa de test se separaron 1160 imágenes; a este grupo de datos se aplicó el método *predict* que otorgó una estimación para cada clase, la estimación estará en el rango de 0 a 1; posteriormente, se aplicará el método *argmax* para que devuelva la clase con mayor estimación. De este modo, solo tendremos 0 (normal) y 1 (neumonía).

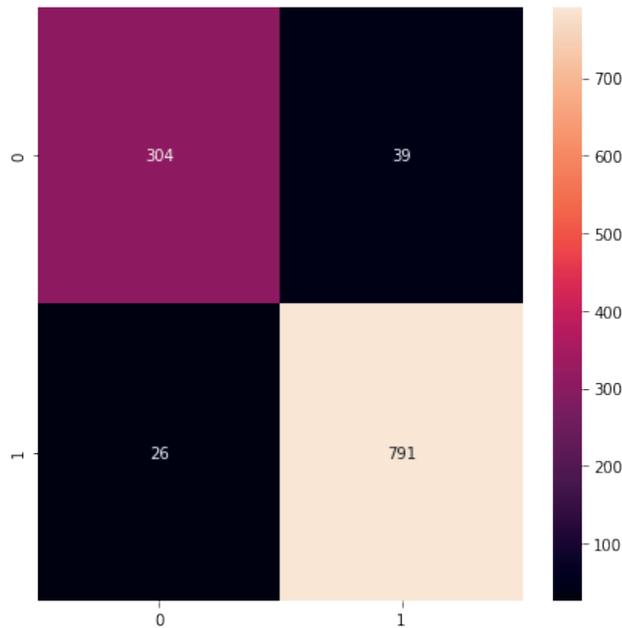


Figura 9. Matriz de confusión para los resultados con Xception, donde 0 = normal y 1 =neumonía

Elaboración propia

En la figura 9 se visualizan los resultados de la matriz de confusión para Xception; se acierta correctamente 304 y 791 veces para “normal” y “neumonía”, respectivamente. No obstante, también obtiene 26 falsos positivos y 39 falsos negativos; es decir, 26 veces predijo que era normal cuando realmente era neumonía y 39 veces predijo que era neumonía cuando era normal.

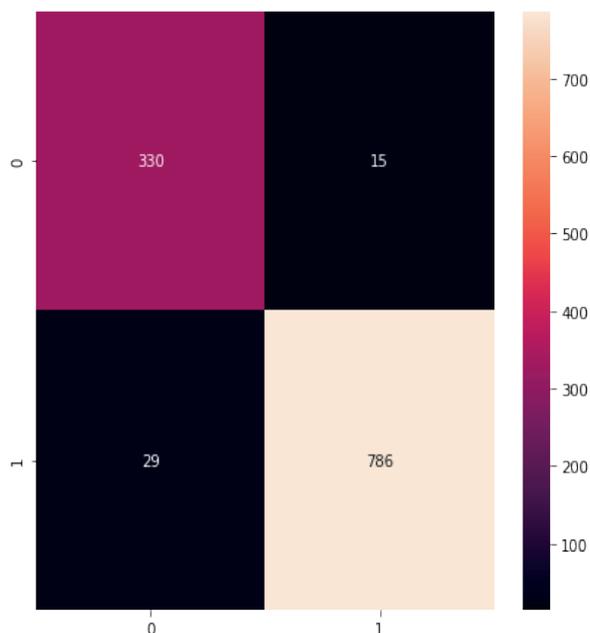


Figura 10. Matriz de confusión para los resultados con MobileNet, donde 0 = normal y 1 = neumonía

Elaboración propia

En la figura 10 se observan los resultados de la matriz de confusión para MobileNet, que acierta correctamente 330 y 786 veces para “normal” y “neumonía” respectivamente. No obstante, también obtiene 29 falsos positivos y 15 falsos negativos; es decir, 29 veces predijo que era normal cuando realmente era neumonía y 15 veces predijo que era neumonía cuando era normal.

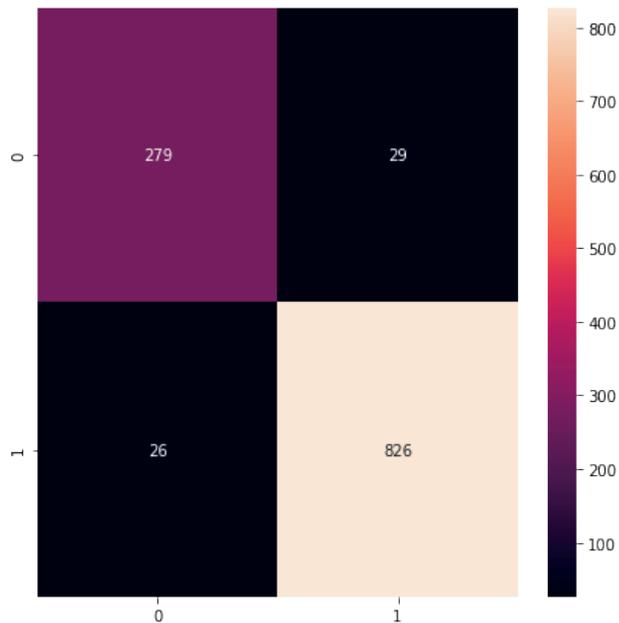


Figura 11. Matriz de confusión para los resultados con InceptionV3, donde 0 = normal y 1 = neumonía
Elaboración propia

En la figura 11 se observan los resultados de la matriz de confusión para InceptionV3, que acierta correctamente 279 y 826 veces para “normal” y “neumonía” respectivamente. No obstante, también obtiene 26 falsos positivos y 29 falsos negativos; es decir, 26 veces predijo que era normal cuando realmente era neumonía y 29 veces predijo que era neumonía cuando era normal.

Tabla 1
Métricas de evaluación para cada algoritmo

Algoritmo	Clase	Accuracy	Precision	Recall	F1-Score
Xception	Normal	0,944	0,92	0,89	0,90
	Neumonía		0,95	0,97	0,96
MobileNet	Normal	0,962	0,92	0,96	0,94
	Neumonía		0,98	0,96	0,97
InceptionV3	Normal	0,953	0,91	0,91	0,91
	Neumonía		0,97	0,97	0,97

Elaboración propia

A partir de la matriz de confusión podemos calcular las métricas de *accuracy*, *precision*, *recall* y F1-Score de cada algoritmo empleado para el presente estudio. Podemos ver los resultados en la tabla 1, cada algoritmo ha demostrado un alto desempeño, pero MobileNet ha obtenido mejores resultados para cada métrica, le sigue InceptionV3, que por poco margen dista de los resultados obtenidos por MobileNet; en el caso de Xception se obtuvieron resultados por debajo de InceptionV3 y MobileNet, esto debido a que sufrió de un sobreajuste más pronunciado en la etapa de entrenamiento como se puede observar en la figura 3.

4. DISCUSIÓN

Los resultados obtenidos son más que aceptables en comparación con otras investigaciones realizadas sobre el mismo *dataset*; en la investigación de Liang y Zheng (2020) se experimentó con VGG16, DenseNet121, InceptionV3, Xception y un modelo propuesto por ellos; el modelo que mejores resultados les proporcionó fue el que proponen, obteniendo un *accuracy* de 90,50 %, en Xception obtuvieron 87,80 % de *accuracy* y con InceptionV3 obtuvieron un 85,30 % de *accuracy*. Mientras que en la presente investigación se obtuvo con MobileNet un 96,20 % de *accuracy*, InceptionV3 obtuvo 95,30 % y Xception obtuvo un 94,40 %. Esta mejora significativa, en comparación de los resultados obtenidos por Liang y Zheng (2020), se debe a la profundidad con las seis capas densas que se añadieron a los tres modelos empleados en la investigación. Al poder extraer y mapear las características de cada imagen del *dataset*, los modelos han podido generalizar adecuadamente, esto se demuestra con los resultados obtenidos.

5. CONCLUSIONES

En este trabajo se presentaron tres algoritmos diferentes preentrenados para la aplicación en la detección de neumonía. Los algoritmos fueron Xception, MobileNet e InceptionV3 que obtuvieron resultados más que aceptables. Xception obtuvo un 94,4 % en la métrica de *accuracy*, 93,5 % en la métrica de *precision*, 93,0 % en la métrica de *recall*, 93,0 % en la métrica de F1-Score. MobileNet obtuvo un 96,2 % en la métrica de *accuracy*, 95,0 % en la métrica de *precision*, 96,0 % en la métrica de *recall*, 95,5 % en la métrica de F1-Score. Por último, InceptionV3 obtuvo un 95,3% en la métrica de *accuracy*, 94,0 % en la métrica de *precision*, 94,0 % en la métrica de *recall*, 94,0 % en la métrica de F1-Score. De los resultados podemos concluir que MobileNet ha demostrado un mejor desempeño ante Xception e InceptionV3 en las diferentes métricas empleadas. Además, su complejidad es muy inferior a los otros algoritmos con 8,1 millones de parámetros, llevando a la utilización de menos recursos computacionales. Los resultados indican que MobileNet cuenta con un nivel muy alto de eficiencia con relación a Xception e InceptionV3.

REFERENCIAS

- Chollet, F. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1800-1807. <https://doi.org/10.1109/CVPR.2017.195>
- Deng, J., Russakovsky, O., Krause, J., Bernstein, M., Berg, A., y Fei-Fei, L. (2014). Scalable Multi-Label Annotation. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*, 3099-3102. <https://doi.org/10.1145/2556288.2557011>
- Hosny, A., Parmar, C., Quackenbush, J., Schwartz, L. H., y Aerts, H. J. W. L. (2018). Artificial Intelligence in Radiology. *Nature Reviews Cancer*, 18(8), 500-510. <https://doi.org/10.1038/s41568-018-0016-5>
- Howard, A., Sandler, M., Chen, B., Wang, W., Chen, L. C., Tan, M., Chu, G., Vasudevan, V., Zhu, Y., Pang, R., Adam, H., y Le, Q. (2019). Searching for MobileNetV3. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 1314-1324. <https://doi.org/10.1109/iccv.2019.00140>
- Kallianos, K., Mongan, J., Antani, S., Henry, T., Taylor, A., Abuya, J., y Kohli, M. (2019). How Far Have We Come? Artificial Intelligence for Chest Radiograph Interpretation. *Clinical Radiology*, 74(5), 338-345. <https://doi.org/10.1016/j.crad.2018.12.015>
- Kermany, D. S., Goldbaum, M., Cai, W., Valentim, C. C., Liang, H., Baxter, S. L., McKeown, A., Yang, G., Wu, X., Yan, F., Dong, J., Prasadha, M. K., Pei, J., Ting, M., Zhu, J., Li, C., Hewett, S., Dong, J., Ziyar, I., ... Zhang, K. (2018). Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning. *Cell*, 172(5), 1122-1131.e9. <https://doi.org/10.1016/j.cell.2018.02.010>
- Liang, G., y Zheng, L. (2020). A Transfer Learning Method with Deep Residual Network for Pediatric Pneumonia Diagnosis. *Computer Methods and Programs in Biomedicine*, 187. <https://doi.org/10.1016/j.cmpb.2019.06.023>
- Liu, N., Wan, L., Zhang, Y., Zhou, T., Huo, H., y Fang, T. (2018). Exploiting Convolutional Neural Networks with Deeply Local Description for Remote Sensing Image Classification. *IEEE Access*, 6, 11215-11227. <https://doi.org/10.1109/ACCESS.2018.2798799>
- WHO. (2019). *Pneumonia*. <https://www.who.int/es/news-room/fact-sheets/detail/pneumonia>
- Song, S., Chaudhuri, K., y Sarwate, A. D. (2013). Stochastic Gradient Descent with Differentially Private Updates. *2013 IEEE Global Conference on Signal and Information Processing*, 245-248. <https://doi.org/10.1109/GlobalSIP.2013.6736861>
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., y Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2818-2826. <https://doi.org/10.1109/CVPR.2016.308>