



DIAGNÓSTICO DE CÁNCER ORAL MEDIANTE ALGORITMOS DE APRENDIZAJE PROFUNDO

DIAGNOSIS OF ORAL CANCER USING DEEP LEARNING ALGORITHMS

Mayra Alejandra Dávila Olivos^{1,*} , Henry Miguel Herrera del Águila¹ ,
Félix Melchor Santos López¹ 

Recibido: 15-05-2023, Recibido tras revisión: 20-05-2024, Aceptado: 28-05-2024, Publicado: 01-07-2024

Resumen

El propósito de este estudio fue diagnosticar automáticamente el cáncer oral en imágenes de labios, mucosa y cavidad oral utilizando aprendizaje profundo. Se propuso un modelo de red neuronal convolucional (CNN) profunda con aumento de datos para el diagnóstico de enfermedades bucodentales. Se desarrolló una CNN profunda de MobileNet para detectar y clasificar la enfermedad de cáncer oral en la zona de los labios, mucosa y cavidad oral. El conjunto de datos de 131 imágenes de labios, mucosa y cavidad oral estaba compuesto por 87 casos positivos y 44 casos negativos. Además, el número de imágenes se multiplicó mediante cambios de corte, enfoque, rotación, brillo y volteo. Se evaluó el rendimiento de diagnóstico de la CNN propuesta a través del cálculo de la exactitud, la precisión, la recuperación, la puntuación F1 y el AUC (área bajo la curva) para la enfermedad de cáncer oral. El rendimiento general del diagnóstico de la enfermedad de cáncer oral alcanzó el 90,9 % de exactitud y 0,91 AUC usando la CNN con el conjunto de datos. El método CNN desarrollado para diagnosticar automáticamente el cáncer oral en imágenes de labios, mucosa y cavidad oral usando aumento de datos mostró una alta exactitud, precisión, recuperación, puntaje F1 y AUC a pesar del número limitado de imágenes de labios, mucosa y cavidad oral utilizadas.

Palabras clave: diagnóstico automático, aumento de datos, cáncer oral, enfermedad bucodental, red neuronal convolucional, salud dental

Abstract

The aim of this study was to use deep learning for the automatic diagnosis of oral cancer, employing images of the lips, mucosa, and oral cavity. A deep convolutional neural network (CNN) model, augmented with data, was proposed to enhance oral cancer diagnosis. We developed a Mobile Net deep CNN designed to detect and classify oral cancer in the lip, mucosa, and oral cavity areas. The dataset comprised 131 images, including 87 positive and 44 negative cases. Additionally, we expanded the dataset by varying cropping, focus, rotation, brightness, and flipping. The diagnostic performance of the proposed CNN was evaluated by calculating accuracy, precision, recall, F1 score, and area under the curve (AUC) for oral cancer. The CNN achieved an overall diagnostic accuracy of 90.9% and an AUC of 0.91 with the dataset for oral cancer. Despite the limited number of images of lips, mucosa, and oral cavity, the CNN method developed for the automatic diagnosis of oral cancer demonstrated high accuracy, precision, recall, F1 score, and AUC when augmented with data.

Keywords: Automatic diagnosis, convolutional neural network, data augmentation, dental health, oral cancer, oral disease

^{1,*}Faculty of Systems Engineering and Informatics, National University of San Marcos, Perú.
Autor para correspondencia ✉: 12200133@unmsm.edu.pe.

Forma sugerida de citación: Dávila Olivos, M. A.; Herrera del Águila, H. M. ; Santos López, F. M. "Diagnóstico de cáncer oral mediante algoritmos de aprendizaje profundo," *Ingenius, Revista de Ciencia y Tecnología*, N.º 32, pp. 58-68, 2024. DOI: <https://doi.org/10.17163/ings.n32.2024.06>.

1. Introducción

Las enfermedades orales representan un desafío significativo para la salud pública global, afectando particularmente a las poblaciones menos privilegiadas debido a su alta prevalencia. Los costos de tratamiento suelen ser prohibitivamente altos y siguen siendo inaccesibles en muchos países de ingresos bajos y medianos. Según la Organización Mundial de la Salud, la gestión de las enfermedades orales se clasifica como la cuarta condición de salud más costosa en las naciones altamente urbanizadas. Dada su profunda repercusión en la salud general, la salud oral es un determinante esencial del bienestar humano y un componente crítico de la atención sanitaria. Además, la presencia de enfermedades orales aumenta el riesgo de padecer condiciones crónicas como diabetes, problemas respiratorios y enfermedades cardiovasculares y cerebrovasculares [1].

El aprendizaje automático (AA), una subcategoría de la inteligencia artificial (IA), emplea técnicas estadísticas, probabilísticas y de optimización que permiten a las máquinas aprender a partir de datos históricos, adquirir información y realizar predicciones sobre nuevos datos basándose en la información aprendida [2,3]. Dentro de la toma de decisiones clínicas dentales, los métodos basados en el aprendizaje profundo (AP), otra subcategoría de la IA, agilizan los procesos y abordan desafíos complejos. Entre estos métodos, una red neuronal convolucional profunda (RNC), un algoritmo bien definido en AP ha demostrado ser altamente efectiva para la segmentación de órganos y la clasificación y detección de órganos y enfermedades en imágenes médicas [4–6].

El aprendizaje automático (AA) ha demostrado una notable precisión y exactitud, superando el juicio humano en la predicción de resultados médicos [2]. Las técnicas de aprendizaje profundo (AP) ofrecen ventajas sobre los métodos basados en características en el análisis de imágenes médicas, superando consistentemente a los profesionales de la salud en la identificación de enfermedades [7].

En el campo del diagnóstico del cáncer oral, el aprendizaje profundo (AP) ha producido resultados prometedores en el análisis automatizado de patología, la obtención de imágenes de la cavidad oral, la imagenología mediante endomicroscopía láser confocal y la imagenología por fluorescencia. Estos avances facilitan la predicción del riesgo de cáncer y los resultados diagnósticos de los pacientes, permitiendo la identificación de patrones sutiles dentro de grandes conjuntos de datos ruidosos. El objetivo final es desarrollar herramientas para mejorar la salud dental pública [2], [7,8].

Este artículo tiene como objetivo implementar un modelo para el diagnóstico del cáncer oral utilizando algoritmos de aprendizaje profundo (AP) de alto rendimiento. El modelo propuesto tiene el poten-

cial de ser un recurso valioso en el proceso de toma de decisiones para el diagnóstico de esta enfermedad.

1.1. Revisión de la literatura

La revisión sistemática de la literatura se llevó a cabo utilizando la metodología PRISMA. Este enfoque facilitó una comprensión integral del trasfondo de la investigación, respaldó este trabajo y demostró la competencia en enfoques de aprendizaje automático (AA) y aprendizaje profundo (AP), asegurando así la relevancia del estudio [9].

El estudio presentado en [2] desarrolla y valida cuatro modelos de aprendizaje automático (AA) para predecir la ocurrencia de metástasis en los ganglios linfáticos en el carcinoma de células escamosas de la lengua oral (OTSCC) en etapa temprana, tanto antes como después de la cirugía. Los modelos de bosque aleatorio y máquina de vectores de soporte muestran un rendimiento predictivo superior en comparación con los métodos tradicionales basados en la profundidad de invasión, la proporción de neutrófilos a linfocitos o la gemación tumoral.

En una investigación separada [4], los científicos automatizan el diagnóstico de quistes y tumores odontogénicos en ambas mandíbulas utilizando radiografías panorámicas. Mejoran una CNN profunda modificada derivada de YOLOv3 para detectar y clasificar estas condiciones. El rendimiento general de la clasificación de enfermedades mejora al utilizar una CNN con un conjunto de datos aumentado en comparación con un conjunto de datos no aumentado.

1.2. Mobile Net

MobileNet utiliza convoluciones separables en profundidad, una técnica que reduce significativamente el número de parámetros en comparación con las redes que utilizan convoluciones regulares con la misma profundidad. Esta reducción de parámetros permite la creación de redes neuronales profundas y ligeras. Desarrollado por Google como una clase de CNN de código abierto, MobileNet es una excelente base para el entrenamiento de clasificadores. Los clasificadores generados con MobileNet no solo son compactos y rápidos, sino que también ayudan a minimizar el tamaño del modelo y los requisitos computacionales. Esto se logra reemplazando los filtros de convolución estándar por convoluciones profundas y puntuales [10].

1.3. Métricas de rendimiento

1. **Curvas de características operativas del receptor (ROC).** Las curvas ROC son representaciones gráficas ampliamente utilizadas para evaluar y comparar el rendimiento de los clasificadores. Estos gráficos bidimensionales ilustran

el compromiso entre la sensibilidad y la especificidad en las predicciones de un clasificador. Demuestran visualmente el rendimiento del clasificador a través de varios umbrales de discriminación, facilitando la clasificación y selección de clasificadores según los requisitos específicos del usuario. Estos requisitos frecuentemente incluyen consideraciones de costos diferenciales de error y demandas de precisión [11].

2. **Área bajo la curva (AUC).** AUC es un valor escalar único que proporciona una medida integral del rendimiento global de un clasificador binario. El valor de AUC varía de 0.5 a 1.0, donde el valor mínimo indica el rendimiento de un clasificador aleatorio, y el valor máximo corresponde al de un clasificador perfecto. En la Figura 1, se presentan las curvas ROC para dos clasificadores de puntuación, A y B. En este ejemplo, el clasificador A tiene un valor de AUC mayor que el clasificador B [12].

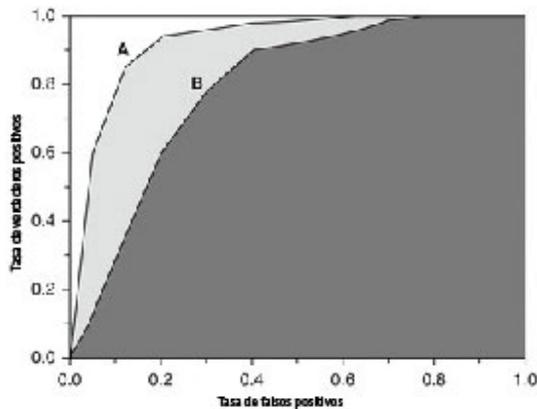


Figura 1. Área bajo la curva [11]

3. **Matriz de confusión.** Una matriz de confusión es una representación tabular que muestra la clase verdadera y la clase predicha de cada caso en el conjunto de pruebas. Esta matriz es esencial para evaluar el rendimiento de un modelo en un problema de clasificación. Al presentar una descomposición clara de verdaderos positivos, verdaderos negativos, falsos positivos y falsos negativos, permite una comprensión detallada del rendimiento del modelo a través de diferentes clases. La Figura 2 proporciona una representación visual de la matriz de confusión, diseñada específicamente para un problema de clasificación binaria [13].

		Valores actuales	
		Positivo	Negativo
Valores predichos	Positivo	VP (Verdadero Positivo)	FP (Falso Positivo)
	Negativo	FN (Falso negativo)	VN (Verdadero negativo)

Figura 2. Matriz de confusión binaria [12]

4. **Exactitud.** Un método de prueba se considera preciso cuando mide con exactitud lo que se pretende medir. En otras palabras, puede determinar de manera efectiva la cantidad o concentración exacta de una sustancia dentro de una muestra [14].

$$Exactitud = \frac{TP + TN}{TP + FP + FN + TN} \dots \quad (1)$$

5. **Precisión.** La precisión en un método de prueba se logra cuando las determinaciones o análisis repetidos sobre la misma muestra arrojan resultados consistentes. En el contexto de la exactitud, un método de prueba preciso exhibe una variación aleatoria mínima, lo que aumenta la confianza en su fiabilidad. La capacidad del método de prueba para reproducir resultados consistentemente a lo largo del tiempo subraya su dependabilidad [14].

$$Precisión = \frac{TP}{TP + FP} \dots \quad (2)$$

6. **Sensibilidad.** Denota la capacidad de una prueba diagnóstica para detectar correctamente a los individuos afectados por una enfermedad o trastorno específico. Una prueba con alta sensibilidad minimiza las instancias de "falsos negativos", en las que la prueba no logra identificar la presencia de una enfermedad a pesar de su existencia real [14].

$$Sensibilidad = \frac{TN}{TN + FP} \dots \quad (3)$$

7. **Puntuación F1.** Integra las mediciones de precisión y sensibilidad en una métrica unificada, facilitando una evaluación comparativa del rendimiento general a través de diversas soluciones. La puntuación F1 opera bajo la suposición de que tanto la precisión como la sensibilidad tienen igual importancia [15].

$$PuntuaciónF1 = 2 * \frac{Precisión * Sensi.}{Precisión + Sensi.} \dots \quad (4)$$

1.4. Cáncer oral

El cáncer oral incluye tumores malignos que afectan el labio, diferentes áreas de la boca y la orofaringe, como se muestra en la Figura 3. Esta forma de cáncer es más común en hombres y personas mayores, con diferencias significativas asociadas con el nivel socioeconómico. Curiosamente, en algunos países de Asia y el Pacífico, el cáncer oral se encuentra entre los tres tipos de cáncer con mayor incidencia [16].



Figura 3. Úlcera de cáncer oral [16]

2. Materiales y métodos

2.1. Adquisición de datos

El conjunto de datos utilizado en este estudio fue adquirido de la plataforma web Kaggle [17], que ofrece acceso abierto a datos descargables. El formato original de los datos consistía en imágenes jpg. El conjunto de datos incluía 131 casos, con 87 presentando imágenes de labios, membranas mucosas y la cavidad oral indicativas de cáncer oral, y los 44 restantes mostrando imágenes sin cáncer oral. La Figura 4 resume visualmente la metodología de investigación, delineando claramente las fases. La implementación de esta etapa se detalla en la Sección 2.6.1.

2.2. Partición del conjunto de datos

La partición del conjunto de datos implica una división no superpuesta de los datos disponibles en dos subconjuntos distintos: el conjunto de datos de entrenamiento y el conjunto de datos de validación. Esta separación proporciona un subconjunto para propósitos analíticos y otro para la verificación del modelo.

1. **Conjunto de entrenamiento.** El conjunto de datos de entrenamiento constituye el 90 % del

conjunto total de datos, abarcando 118 imágenes de labios, mucosa y cavidad oral. Esto incluye 78 imágenes indicativas de cáncer oral y 40 imágenes sin signos de cáncer oral.

2. **Conjunto de validación.** El conjunto de datos de validación comprende el 10 % del conjunto total de datos, con 13 imágenes de labios, mucosa y cavidad oral. Esto incluye 9 imágenes indicativas de cáncer oral y 4 imágenes sin signos de cáncer oral.

La implementación de esta etapa se detalla en la sección 2.6.2.

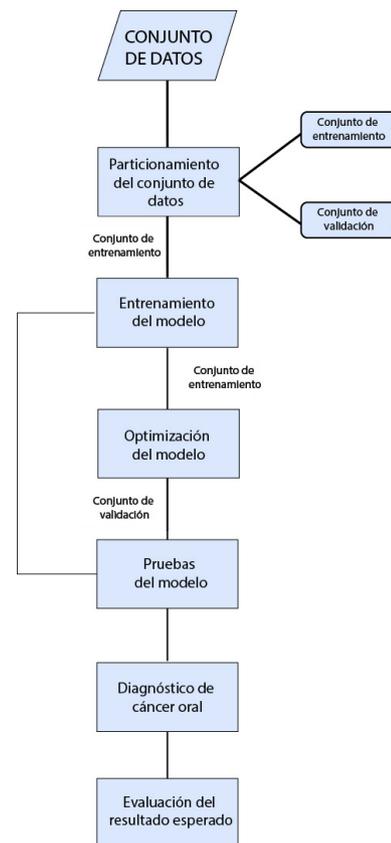


Figura 4. Metodología de desarrollo del modelo

2.3. Entrenamiento, optimización y pruebas del modelo

El modelo de red neuronal convolucional (RNC) propuesto en este estudio fue implementado y entrenado en la plataforma Kaggle, utilizando Python como lenguaje de programación, debido a sus extensas capacidades en aprendizaje automático (AA) y aprendizaje profundo (AP). Estas características hacen que Python sea particularmente adecuado para gestionar las complejidades de la tarea. Durante esta fase, el conjunto de datos de entrenamiento sirvió no solo para el entrenamiento inicial, sino también para la prevalidación

del modelo, sentando así las bases para una mayor optimización y pruebas.

Al validar el modelo desarrollado, que exhibe métricas de rendimiento que oscilan entre 0,5 y 1,0, se identifican las áreas que requieren mejora para optimizar su rendimiento. Se realizan ajustes utilizando el conjunto de datos de entrenamiento. Una vez que se logran métricas de evaluación satisfactorias, el modelo se somete a pruebas para verificar su efectividad y fiabilidad.

Para evaluar el modelo propuesto, se emplea el conjunto de datos de validación para confirmar su alto rendimiento. Si los resultados difieren de las expectativas, se realizan iteraciones adicionales de entrenamiento, optimización y pruebas hasta que se logren los resultados deseados. Este proceso iterativo se documenta en las secciones 2.6.3 y 2.6.4.

2.4. Diagnóstico de la enfermedad del cáncer oral

Basándose en los resultados de rendimiento obtenidos de las pruebas del modelo, se determina la capacidad para el diagnóstico automático del cáncer oral. Este diagnóstico se refiere a las imágenes de los labios, mucosa y cavidad oral utilizadas en el modelo. La implementación de esta etapa se detalla en la sección 2.6.5.

2.5. Evaluación del resultado esperado

Después del diagnóstico de cáncer oral, los resultados obtenidos se evalúan comparando su exactitud, precisión, sensibilidad y puntuación F1. A través de esta comparación, se determina que el diagnóstico proporcionado por el modelo propuesto arroja resultados satisfactorios. La implementación de esta etapa se detalla en la sección 2.6.6.

2.6. Implementación

2.6.1. Carga de bibliotecas y lectura de datos

El desarrollo de la solución comienza con la carga de bibliotecas esenciales, como matplotlib, NumPy y pandas. Se definen parámetros globales y se obtienen imágenes de los labios, mucosa y cavidad oral, ya sean indicativas de la enfermedad o no. Los parámetros clave incluyen:

1. **Tamaño:** Tamaño de entrada [18].
2. **Épocas:** El número de iteraciones sobre el conjunto de datos completo [19].
3. **Tamaño del lote:** División del conjunto de datos en múltiples lotes más pequeños [19].
4. **Pliegues:** El número de pliegues en los que se dividirá el conjunto de datos [20].

2.6.2. Generación del conjunto de datos

Se crea un conjunto de datos que comprende imágenes de labios, membranas mucosas y la cavidad oral, con o sin la enfermedad. Las imágenes se redimensionan y su cantidad se aumenta mediante diversas alteraciones, incluyendo recorte, ajuste de enfoque, rotación, modificación de brillo y volteo.

El conjunto de datos, inicialmente desequilibrado con 44 casos sin cáncer oral y 87 casos con cáncer oral, se somete a un balanceo de clases. Se determinan las clases para identificar la presencia o ausencia de la enfermedad, y las imágenes se segmentan según si muestran o no cáncer oral.

2.6.3. Creación de un modelo

El modelo de RNC propuesto se establece utilizando "MobileNet" como la arquitectura elegida. Además, se emplean los siguientes atributos:

1. **Early Stopping.** Configurado con una paciencia de "10" ciclos de entrenamiento, este atributo monitorea una métrica específica para detectar cualquier signo de mejora antes de concluir [21].
2. **Adam.** Este optimizador implementa el algoritmo de Adam, un método de descenso de gradiente estocástico, basado en la estimación adaptativa de momentos de primer y segundo orden [22].
3. **Sequential.** Este atributo proporciona funciones de entrenamiento e inferencia para el modelo [23].
4. **Conv2D.** Una capa de convolución 2D que genera un kernel de convolución que se aplica sobre las capas de entrada, produciendo un tensor de salida [24].
5. **Relu.** Aplicado para activar la función de activación de unidad lineal rectificadora [25].
6. **MaxPooling2D.** Este atributo realiza una operación de pooling máximo para datos espaciales 2D [26].
7. **Flatten.** Utilizado para aplanar la entrada sin afectar el tamaño del lote [27].
8. **Dense.** Esta capa aplica pesos a todos los nodos de la capa precedente [28].
9. **Dropout.** Durante el entrenamiento, este atributo configura aleatoriamente unidades de entrada a 0 con una frecuencia especificada en cada paso, ayudando a prevenir el sobreajuste [29].
10. **SoftMax.** Convierte un vector de valores en una distribución de probabilidad [25].
11. **Compile.** Un método que acepta un argumento de métrica y una lista de métricas [30].

12. **Categorical cross entropy.** Este atributo calcula la pérdida de entropía cruzada entre etiquetas y predicciones [31].

13. **Accuracy.** Calcula la frecuencia con la que las predicciones coinciden con las etiquetas [32].

2.6.4. Entrenamiento con K-fold

El entrenamiento del modelo se ejecuta utilizando los siguientes atributos:

1. **K-fold.** El conjunto de datos se divide en K pliegues, donde cada pliegue sirve como el conjunto de prueba, mientras que el resto del conjunto de datos se utiliza como conjunto de entrenamiento [33, 34].
2. **Stratified Fold.** Este atributo asegura una mayor validación cruzada, preservando la distribución de clases en el conjunto de datos, tanto en las divisiones de entrenamiento como de validación [35].
3. **To categorical.** Convierte un vector de clases (enteros) en una matriz de clases binarias [36].
4. **Image Data Generator.** Este atributo facilita la generación de bloques de entrenamiento y realiza el aumento de datos, incrementando el número de imágenes mediante modificaciones como *zoom*, *escalado*, *volteo horizontal*, etc. [37].
5. **Fit.** Esta función se utiliza para entrenar el modelo durante un número fijo de épocas (iteraciones sobre un conjunto de datos) [38].
6. **Model Checkpoint.** Sirve como un *callback* para guardar el modelo o los pesos del modelo de Keras en intervalos especificados [39].
7. **Create model.** Esta función es responsable de crear y entrenar una nueva instancia del modelo [40].

De manera similar, se propone otra RNC para crear el modelo, utilizando “ResNet152V2”, “DenseNet121” y “EfficientNetB6” como las arquitecturas elegidas. La Tabla 1 ilustra una comparación del rendimiento del modelo utilizando estas arquitecturas de aprendizaje profundo.

Tabla 1. Comparación de arquitecturas de aprendizaje profundo

ResNet152V2	Precisión (%)	Recall (%)	Puntuación F1 (%)
Con cáncer oral	90	93	92
Sin cáncer oral	85	80	82
Precisión AUC	0.8855		
	0.863244514106583		
DenseNet121	Precisión (%)	Recall (%)	Puntuación F1 (%)
Con cáncer oral	85	94	90
Sin cáncer oral	86	68	76
Precisión AUC	0.8550		
	0.812173458725183		

Con respecto al uso de “Efficient Net B6”, surge un problema de memoria al comienzo del pliegue 3 durante el entrenamiento del modelo, lo que impide la finalización del proceso.

De manera similar, aunque los resultados del modelo utilizando las arquitecturas “Res Net 152V2”, “Dense Net 121” y “Mobile Net”, en el mismo conjunto de datos, son comparables, el número de hiperparámetros utilizados en MobileNet es menor. Las figuras 5, 6, y 7 ilustran el número de hiperparámetros obtenidos para cada arquitectura:

“ResNet152V2” utilizó 76MM, como se muestra en la Figura 5.

Layer (type)	Output Shape	Param #
resnet152v2 (Functional)	(None, 7, 7, 2048)	58331648
conv2d (Conv2D)	(None, 5, 5, 64)	1179712
conv2d_1 (Conv2D)	(None, 3, 3, 64)	36928
max_pooling2d_3 (MaxPooling2D)	(None, 1, 1, 64)	0
flatten (Flatten)	(None, 64)	0
dense (Dense)	(None, 4096)	266240
dropout (Dropout)	(None, 4096)	0
dense_1 (Dense)	(None, 4096)	16781312
dropout_1 (Dropout)	(None, 4096)	0
dense_2 (Dense)	(None, 2)	8194
Total params: 76,604,034		
Trainable params: 76,460,290		

Figura 5. Número de hiperparámetros de ResNet152V2

“DenseNet121” utilizó 24MM, como se muestra en la Figura 6.

Layer (type)	Output Shape	Param #
densenet121 (Functional)	(None, 7, 7, 1024)	7037504
conv2d (Conv2D)	(None, 5, 5, 64)	589888
conv2d_1 (Conv2D)	(None, 3, 3, 64)	36928
max_pooling2d (MaxPooling2D)	(None, 1, 1, 64)	0
flatten (Flatten)	(None, 64)	0
dense (Dense)	(None, 4096)	266240
dropout (Dropout)	(None, 4096)	0
dense_1 (Dense)	(None, 4096)	16781312
dropout_1 (Dropout)	(None, 4096)	0
dense_2 (Dense)	(None, 2)	8194

Total params: 24,720,066
Trainable params: 24,636,418

Figura 6. Número de hiperparámetros de DenseNet121

"MobileNet" utilizó 24MM, como se muestra en la Figura 7.

Layer (type)	Output Shape	Param #
mobilenet_1.00_224 (Function)	(None, 7, 7, 1024)	3228864
conv2d_52 (Conv2D)	(None, 5, 5, 64)	589888
conv2d_53 (Conv2D)	(None, 3, 3, 64)	36928
max_pooling2d_26 (MaxPooling)	(None, 1, 1, 64)	0
flatten_26 (Flatten)	(None, 64)	0
dense_84 (Dense)	(None, 4096)	266240
dropout_58 (Dropout)	(None, 4096)	0
dense_85 (Dense)	(None, 4096)	16781312
dropout_59 (Dropout)	(None, 4096)	0
dense_86 (Dense)	(None, 2)	8194

Total params: 20,911,426
Trainable params: 20,889,538

Figura 7. Número de hiperparámetros de MobileNet

2.6.5. Verificación del modelo

El proceso de verificación del modelo se realiza utilizando todo el conjunto de datos de validación, empleando los siguientes atributos:

1. **Evaluate.** Esta función devuelve el valor de pérdida y los valores métricos del modelo en modo de prueba [38].
2. **Predict.** Genera predicciones de salida para las muestras de entrada [38].
3. **Confusion matrix.** El cálculo de la matriz de confusión se utiliza para evaluar la precisión de una clasificación [41].
4. **Subplot.** Este atributo obtiene la posición del índice en una cuadrícula con "n" filas y "n" columnas [42].
5. **Heatmap.** Se utiliza para obtener un mapa de calor de activación de clases para un modelo de clasificación de imágenes [43].
6. **Set-ticklabels.** Esta función establece los nombres de destino para la matriz de confusión.
7. **Roc curve.** Este atributo calcula la curva ROC [44].
8. **Roc-auc-core.** Calcula el AUC de la curva ROC a partir de las puntuaciones de predicción [45].

2.6.6. Trazado de curvas AUC

La Figura 8 muestra un gráfico de la tasa de verdaderos positivos frente a la tasa de falsos positivos, ilustrando el AUC a través de líneas. Esta visualización permite observar la relación entre estas dos variables.

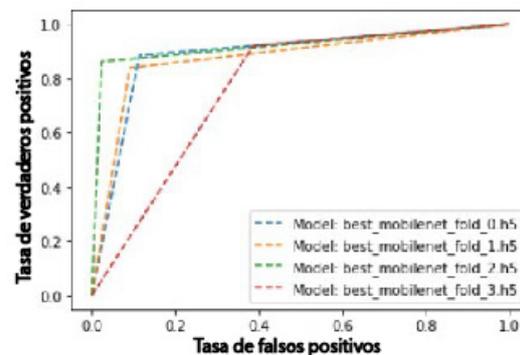


Figura 8. Trazado de curvas AUC

Se confirma que el modelo de RNC propuesto demuestra un alto rendimiento en la clasificación de la presencia y ausencia de cáncer oral.

3. Resultados y discusión

La Figura 9 ilustra el gráfico de los valores de precisión frente al número de épocas, utilizando líneas para visualizar la relación entre estas dos variables.

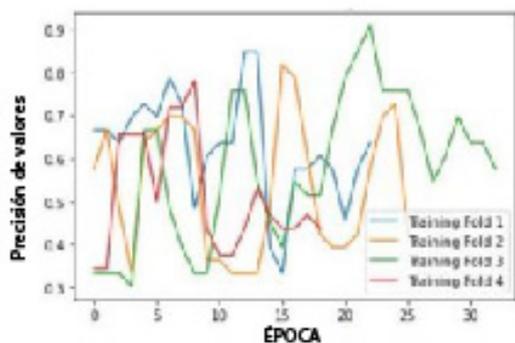


Figura 9. Precisión del modelo por pliegue

Al entrenar el modelo, se obtiene la precisión para cada pliegue.

En el primer entrenamiento, con el pliegue 1, se logra una "precisión de valor" de 0,84848.

En el segundo entrenamiento, con el pliegue 2, se logra una "precisión de valor" de 0,81818.

En el tercer entrenamiento, con el pliegue 3, se logra una "precisión de valor" de 0,90909.

En el cuarto entrenamiento, con el pliegue 4, se logra una "precisión de valor" de 0,78125.

Se observa que el pliegue 1 alcanza una buena "precisión de valor"; el pliegue 2 disminuye la "precisión de valor"; el pliegue 3 alcanza la mayor "precisión de valor" con un valor de 0,90909, y el pliegue 4 disminuye la "precisión de valor".

Durante la validación del modelo, se obtienen las siguientes métricas para el modelo guardado como "best mobilenet fold 0.h5," como se ilustra en la Tabla 2.

Tabla 2. Métricas de rendimiento del modelo simple con la partición 1

	Precisión (%)	Recall (%)	Puntuación F1 (%)
Con cáncer oral	80	89	84
Sin cáncer oral	94	89	91
AUC	0.8857105538140021		

El archivo "best mobilenet fold 1.h5" alcanza las siguientes clasificaciones, como se muestra en la Tabla 3.

Tabla 3. Métricas de rendimiento del modelo simple con la partición 2

	Precisión (%)	Recall (%)	Puntuación F1 (%)
Con cáncer oral	74	91	82
Sin cáncer oral	95	84	89
AUC	0.8740856844305119		

El archivo "best mobilenet fold 2.h5" alcanza las siguientes clasificaciones, como se muestra en la Tabla 4.

Tabla 4. Métricas de rendimiento del modelo simple con la partición 3

	Precisión (%)	Recall (%)	Puntuación F1 (%)
Con cáncer oral	78	98	87
Sin cáncer oral	99	86	92
AUC	0.9196708463949843		

El archivo "best mobilenet fold 3.h5" alcanza las siguientes clasificaciones, como se muestra en la Tabla 5.

Tabla 5. Métricas de rendimiento del modelo simple con la partición 4

	Precisión (%)	Recall (%)	Puntuación F1 (%)
Con cáncer oral	79	61	69
Sin cáncer oral	82	92	87
AUC	0.7665882967607105		

Se concluye que el modelo "best mobilenet fold 2.h5" se destaca como la opción óptima, exhibiendo la mayor precisión (78 % y 99 %), sensibilidad (98 % y 86 %), puntuación F1 (87 % y 92 %) y AUC (0,9196708463949843) tanto para casos no cancerosos como cancerosos, superando el rendimiento de otros modelos.

Comparativamente, entre las arquitecturas "ResNet152V2", "DenseNet121" y "MobileNet", se demuestra que la arquitectura "MobileNet" es óptima en términos de optimización de recursos, utilizando veinte millones de hiperparámetros, mientras que las arquitecturas "ResNet152V2" y "DenseNet121" utilizan un número significativamente mayor de hiperparámetros.

Además de las métricas de rendimiento de las arquitecturas "ResNet152V2", "DenseNet121" y "EfficientNetB6" evaluadas, el modelo presentado en [7] demostró una precisión de 84,3 %, una sensibilidad de 83,0 %, una puntuación F1 de 83,6 % y un AUC de 0,8974. En contraste, el modelo propuesto en este estudio mostró mejoras, logrando una precisión de 88,5 %, una sensibilidad de 92,0 %, una puntuación F1 de 89,5 % y un AUC de 0,9196708463949843. Por lo tanto, se observa una mejora notable en el rendimiento general, que oscila entre el 2 % y el 9 %.

4. Conclusiones

Este estudio resalta el potencial de la IA para abordar problemas de salud oral, particularmente el cáncer oral, que afecta a una parte significativa de la población. La investigación enfatiza la efectividad del aprendizaje profundo (AP) y concluye que las redes neuronales convolucionales (RNC) son un algoritmo adecuado de AP para procesar imágenes de la mucosa y la cavidad oral. Las RNC toman estas imágenes como entrada y asignan pesos a elementos específicos para distinguir entre ellos. La elección de la RNC MobileNet se

justifica por su capacidad para reducir el tamaño del modelo y la computación al reemplazar los filtros de convolución estándar con convoluciones profundas y puntuales.

El estudio utiliza la plataforma de código abierto Kaggle e implementa el modelo utilizando el lenguaje de programación Python. La evaluación de varias métricas de rendimiento arroja una precisión de 0,90909, confirmando que el modelo de RNC propuesto demuestra un alto rendimiento diagnóstico para el cáncer oral. En cuanto a la cantidad de imágenes, el estudio especifica que el uso de más imágenes mejora los resultados del modelo de AP propuesto. Además, la evaluación de diferentes arquitecturas de RNC ayuda a comprender su rendimiento, facilitando la determinación del modelo más óptimo.

En última instancia, esta investigación afirma que el modelo desarrollado está listo para su aplicación práctica, ofreciendo un valioso apoyo para la toma de decisiones dentales en escenarios de diagnóstico en tiempo real.

4.1. Trabajo futuro

Los esfuerzos continuados en la recolección de más imágenes de labios, mucosa y cavidad oral que representen diversas condiciones orales, incluido el cáncer oral, serán un punto focal para estudios futuros. Se cree que aumentar el conjunto de datos de imágenes y colaborar con instituciones clínicas públicas y privadas para su evaluación mejorará significativamente los resultados y facilitará la aplicación práctica del modelo. Reconociendo el papel crucial de los grandes conjuntos de datos en la optimización de los algoritmos de aprendizaje profundo (AP), los resultados actuales son prometedores y sirven como un primer paso para avanzar en esta línea de investigación. Además, los próximos esfuerzos de investigación se centrarán en evaluar el rendimiento del método de RNC propuesto en el diagnóstico de un espectro más amplio de enfermedades orales.

Referencias

- [1] L. A. Zanella-Calzada, C. E. Galván-Tejada, N. M. Chávez-Lamas, J. Rivas-Gutiérrez, R. Magallanes-Quintanar, J. M. Celaya-Padilla, J. I. Galván-Tejada, and H. Gamboa-Rosales, “Deep artificial neural networks for the diagnostic of caries using socioeconomic and nutritional features as determinants: Data from NHANES 2013–2014,” *Bioengineering*, vol. 5, no. 2, 2018. [Online]. Available: <https://doi.org/10.3390/bioengineering5020047>
- [2] J. Shan, R. Jiang, X. Chen, Y. Zhong, W. Zhang, L. Xie, J. Cheng, and H. Jiang, “Machine learning predicts lymph node metastasis in early-stage oral tongue squamous cell carcinoma,” *Journal of Oral and Maxillofacial Surgery*, vol. 78, no. 12, pp. 2208–2218, 2020. [Online]. Available: <https://doi.org/10.1016/j.joms.2020.06.015>
- [3] A. M. Bur, A. Holcomb, S. Goodwin, J. Woodroof, O. Karadaghy, Y. Shnayder, K. Kakarala, J. Brant, and M. Shew, “Machine learning to predict occult nodal metastasis in early oral squamous cell carcinoma,” *Oral Oncology*, vol. 92, pp. 20–25, 2019. [Online]. Available: <https://doi.org/10.1016/j.oraloncology.2019.03.011>
- [4] O. Kwon, T.-H. Yong, S.-R. Kang, J.-E. Kim, K.-H. Huh, M.-S. Heo, S.-S. Lee, S.-C. Choi, and W.-J. Yi, “Automatic diagnosis for cysts and tumors of both jaws on panoramic radiographs using a deep convolution neural network,” *Dentomaxillofacial Radiology*, vol. 49, no. 8, p. 20200185, Dec 2020. [Online]. Available: <https://doi.org/10.1259/dmfr.20200185>
- [5] X. Zhang, Y. Liang, W. Li, C. Liu, D. Gu, W. Sun, and L. Miao, “Development and evaluation of deep learning for screening dental caries from oral photographs,” *Oral Diseases*, vol. 28, no. 1, pp. 173–181, 2022. [Online]. Available: <https://doi.org/10.1111/odi.13735>
- [6] H.-J. Chang, S.-J. Lee, T.-H. Yong, N.-Y. Shin, B.-G. Jang, J.-E. Kim, K.-H. Huh, S.-S. Lee, M.-S. Heo, S.-C. Choi, T.-I. Kim, and W.-J. Yi, “Deep learning hybrid method to automatically diagnose periodontal bone loss and stage periodontitis,” *Scientific Reports*, vol. 10, no. 1, p. 7531, May 2020. [Online]. Available: <https://doi.org/10.1038/s41598-020-64509-z>
- [7] H. Lin, H. Chen, L. Weng, J. Shao, and J. Lin, “Automatic detection of oral cancer in smartphone-based images using deep learning for early diagnosis,” *Journal of Biomedical Optics*, vol. 26, no. 8, p. 086007, 2021. [Online]. Available: <https://doi.org/10.1117/1.JBO.26.8.086007>
- [8] W. Li, Y. Liang, X. Zhang, C. Liu, L. He, L. Miao, and W. Sun, “A deep learning approach to automatic gingivitis screening based on classification and localization in RGB photos,” *Scientific Reports*, vol. 11, no. 1, p. 16831, Aug 2021. [Online]. Available: <https://doi.org/10.1038/s41598-021-96091-3>
- [9] M. A. Dávila Olivos and F. M. Santos López, “Prediction models of oral diseases: A systematic review of the literature,” in *Emerging Research in Intelligent Systems*, G. F. Olmedo Cifuentes, D. G. Arcos Avilés, and H. V. Lara Padilla,

- Eds. Cham: Springer Nature Switzerland, 2024, pp. 309–322. [Online]. Available: https://doi.org/10.1007/978-3-031-52255-0_22
- [10] A. Pujara, “Image classification with mobilenet,” *Analytics Vidhya*, 2020. [Online]. Available: <https://n9.cl/coutpg>
- [11] F. Melo, *Receiver Operating Characteristic (ROC) Curve*. New York, NY: Springer New York, 2013, pp. 1818–1823. [Online]. Available: https://doi.org/10.1007/978-1-4419-9863-7_242
- [12] —, *Area under the ROC Curve*. New York, NY: Springer New York, 2013, pp. 38–39. [Online]. Available: https://doi.org/10.1007/978-1-4419-9863-7_209
- [13] H. Rhys, *Machine Learning with R, the tidyverse, and mlr*. Manning Publications, 2020. [Online]. Available: <https://n9.cl/q3hijw>
- [14] LabTests Online UK. (2018) Accuracy, precision, specificity & sensitivity. Association for Laboratory Medicine. [Online]. Available: <https://n9.cl/8cvyq>
- [15] J. Martínez Heras. (2018) Machine learning lectures esa. GitHub, Inc. [Online]. Available: <https://n9.cl/7yim6>
- [16] OMS. (2021) Salud bucodental. Organización Mundial de la Salud. [Online]. Available: <https://n9.cl/zpz0f>
- [17] Kaggle. (2020) Oral cancer (lips and tongue) images. Kaggle. [Online]. Available: <https://n9.cl/7ftbq>
- [18] Keras. (2022) Mobilenet, mobilenetv2, and mobilenetv3. Keras. [Online]. Available: <https://n9.cl/dcv52>
- [19] S. Sharma. (2022) Epoch vs batch size vs iterations. Medium. [Online]. Available: <https://n9.cl/wlxncj>
- [20] S. Manna. (2022) K-fold cross validation for deep learning models using keras. Medium. [Online]. Available: <https://n9.cl/hmyvr>
- [21] Keras. (2022) Earlystopping. Keras. [Online]. Available: <https://n9.cl/undx7>
- [22] —. (2022) Adam. Keras. [Online]. Available: <https://n9.cl/x9m53>
- [23] —. (2022) The sequential class. Keras. [Online]. Available: <https://n9.cl/yi56j>
- [24] GeeksforGeeks. (2022) Keras.conv2d class. Geeks for Geeks. [Online]. Available: <https://n9.cl/6bemi>
- [25] Keras. (2022) Layer activation functions. Keras. [Online]. Available: <https://n9.cl/d9yeb>
- [26] —. (2022) Maxpooling2d layer. Keras. [Online]. Available: <https://n9.cl/51sbk>
- [27] —. (2022) Flatten layer. Keras. [Online]. Available: <https://n9.cl/cufk4>
- [28] I. Hull, *Dense layers – Introduction to tensorflow in Python*. DataCamp. [Online]. Available: <https://n9.cl/hny28>
- [29] Keras. (2022) Dropout layer. Keras. [Online]. Available: <https://n9.cl/02hdv>
- [30] —. (2022) Metrics. Keras. [Online]. Available: <https://n9.cl/jmihj>
- [31] —. (2022) Probabilistic metrics. Keras. [Online]. Available: <https://n9.cl/b3w2a>
- [32] —. (2022) Accuracy metrics. Keras. [Online]. Available: <https://n9.cl/713dt>
- [33] Data Science Team. (2020) Validación cruzada k-fold. Data Science Team. [Online]. Available: <https://n9.cl/c2i0bp>
- [34] R. Delgado. (2018) Introducción a la validación cruzada (k-fold cross validation) en R. Amazon-aws. [Online]. Available: <https://n9.cl/ijyq>
- [35] S. Yildirim. (2020) How to train test split : Kfold vs stratifiedkfold. Medium. [Online]. Available: <https://n9.cl/ymp9q>
- [36] Keras. (2022) Python & numpy utilities. Keras. [Online]. Available: <https://n9.cl/zrghg>
- [37] J. Utrera Burgal. (2019) Tratamiento de imágenes usando imagedatagenerator en keras. Knowmad mood. [Online]. Available: <https://n9.cl/5gobr>
- [38] Keras. (2022) Model training apis. Keras. [Online]. Available: <https://n9.cl/4gjr6>
- [39] —. (2022) Modelcheckpoint. Keras. [Online]. Available: <https://n9.cl/wvut7>
- [40] Tensorflow. (2022) Guardar y cargar modelos. TensorFlow. [Online]. Available: <https://n9.cl/cjflnu>
- [41] Scikit Learn. (2022) sklearn.metrics.confusion matrix. Scikit-learn developers. [Online]. Available: <https://n9.cl/ya6b1h>
- [42] Matplotlib. (2022) matplotlib.pyplot.subplot. Matplotlib. [Online]. Available: <https://n9.cl/vlv1fe>

- [43] Keras. (2022) Grad-cam class activation visualization. Keras. [Online]. Available: <https://n9.cl/r5l7k>
- [44] Scikitlearn. (2022) sklearn.metrics.roc curve. Scikitlearn. [Online]. Available: <https://n9.cl/qles5>
- [45] ——. (2022) sklearn.metrics.roc auc score. Scikitlearn. [Online]. Available: <https://n9.cl/1zf6r>