



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES CUAUTILÁN

Clasificación de patologías en radiografías de tórax
basada en redes neuronales

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

LICENCIADO EN TECNOLOGÍA

P R E S E N T A :

MARIO ROSAS OTERO

ASESOR

M. EN C. BERENICE MONTALVO LEZAMA

COASESOR

M. EN C. RICARDO MONTALVO LEZAMA

ESTADO DE MÉXICO, 2023



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE ESTUDIOS SUPERIORES
CUAUTITLÁN



**Clasificación de patologías en radiografías de
tórax basada en redes neuronales**

T E S I S

Que para optar por el grado de:

Licenciado en Tecnología

Presenta:

Mario Rosas Otero

Tutores:

M. en C. Berenice Montalvo Lezama

M. en C. Ricardo Montalvo Lezama

PCIC, IIMAS, UNAM.

Estado de México

Agradecimientos

Son pocas las palabras para este espacio y con ellas poder agradecer a cada persona y vivencia que me han permitido llegar a este momento. Estoy muy agradecido, me siento satisfecho y orgulloso de redactar este texto para expresar mi gratitud a todas y cada una de las personas que han estado en el caminar de mi vida, sin importar si nuestro encuentro fue fugaz o ha sido permanente, de igual forma han dejado en mí una experiencia que ha influido para formarme como hijo, amigo, alumno y como ser humano.

La Universidad Nacional Autónoma de México me concedió las herramientas para trazar y construir los cimientos de mi proyecto de vida e iniciar el camino. A la Facultad de Estudios Superiores Cuautitlán agradezco que me permitió tener las facilidades necesarias para concluir mi licenciatura. Agradezco a los profesores que me han compartido conocimiento, sabiduría y experiencia; en especial al profesor David T. quien siempre ha estado al pendiente de mi avance y mi formación académica, me ha brindado consejos y oportunidades. Mis compañeros en la licenciatura, todos: Armando, Sergio, Darío, Emmanuel... Y en especial Rubén M. y Natalia S. quienes además de compañeros se han convertido en mis hermanos, todos han sido un pilar muy importante en este camino y han hecho de esto, una experiencia inolvidable.

Agradezco profundamente a mis asesores Ricardo y Berenice, quienes además de instruirme académicamente han apoyado mi interés por seguir aprendiendo y desarrollándome personal y profesionalmente. Desde el primer curso en el que tuve la oportunidad de conocerlos, durante el desarrollo de este proyecto y hasta este momento son un ejemplo e inspiración para continuar mi camino profesional.

Recuerdo con satisfacción a las personas que conocí y con las que interactué en todos los eventos extracurriculares en los que participé durante la licenciatura, representaron una gran motivación para poner todo mi esfuerzo en seguir adelante. Especialmente, agradezco a las personas de la RIIAA, compartir con ellos ha sido grandioso.

Mi familia como uno de los ejes centrales en mi vida me han formado, educado y brindado todo lo necesario para poder cumplir con esta etapa y todas las que le anteceden. Es claro para mí que sin ellos esto no sería posible; agradezco a mis papás, abuelos, tíos, primos y sobrinos. Mi padre, quien siempre ha estado ahí para mí, y me ha brindado su apoyo incondicional para cumplir todas mis metas. Mi madre es y será siempre un motor, una inspiración y un amor por el cual seguir adelante, cada día ella me enseña y motiva a ser una mejor persona, a superarme día con día.

“Un scientifique dans son laboratoire est non seulement un technicien : il est aussi un enfant placé devant des phénomènes naturels qui l'impressionnent comme des contes de fées.”

Marie Curie

Índice

1	Introducción	1
1.1	Contexto	1
1.2	Problemática	2
1.3	Motivación	2
1.4	Objetivos	3
2	Marco teórico	4
2.1	Tipos de aprendizaje	5
2.2	Tareas de clasificación en imágenes	6
2.3	Redes neuronales artificiales	7
2.3.1	Neurona artificial	7
2.3.2	Perceptrón multicapa	8
2.3.3	Retropropagación	10
2.4	Redes convolucionales	11
2.4.1	Representación de imágenes	11
2.4.2	Capas de convolución	12
2.4.3	Capas de submuestreo	13
2.4.4	Normalización	14
2.4.5	Arquitectura general	15
2.5	Arquitecturas notables	16
2.6	Transferencia de conocimiento	17
3	Conjunto de datos ChestX-ray8	19
3.1	Motivación de uso	20
3.2	Versiones reducida y completa	20
3.3	Análisis exploratorio de los datos	20
3.3.1	Estadísticas de la versión completa	21
3.3.2	Estadísticas de la versión reducida	24
3.3.3	Ejemplos del conjunto de datos	27
4	Metodología	31
4.1	Conjunto de datos	31
4.1.1	Partición	31
4.1.2	Procesamiento	32
4.1.3	Aumentado de datos	32
4.2	Esquema de experimentación	33
4.3	Evaluación	34
4.3.1	Curva ROC	34

4.3.2	Métricas multietiqueta	35
4.4	Recursos de hardware y software	36
5	Análisis de resultados	37
5.1	Aumentado de datos	37
5.2	Arquitectura	39
5.3	Preentrenamiento	41
5.4	Versiones reducida y completa	43
5.5	Configuración final	43
6	Conclusiones	47
6.1	Conclusiones	47
6.2	Trabajo a futuro	47
7	Referencias	49

Índice de figuras

1	Relación entre inteligencia artificial, aprendizaje de máquinas y aprendizaje profundo.	4
2	Tipos de aprendizaje en aprendizaje de máquinas.	5
3	Problemas de clasificación.	6
4	Ciclo de aprendizaje supervisado.	7
5	Neurona artificial.	8
6	Perceptrón multicapa con dos capas ocultas.	9
7	Perceptrón multicapa con una sola capa oculta y una neurona de salida.	9
8	Etapas de procesamiento de una red neuronal convolucional. Primero, se toma como entrada una imagen y obtienen sus representaciones por la etapa de extracción de características. Después, en la etapa de clasificación, un perceptrón multicapa discrimina las representaciones en las categorías de interés.	11
9	Procesamiento de una capa de convolución para una entrada de 5×5 con dos filtros de 3×3 , paso 2 y relleno de 1.	12
10	Procesamiento de una capa de submuestreo máximo para una entrada de 5×5 con paso de 1.	14
11	Dimensiones en las que opera la normalización por lote.	15
12	Patrón genérico de construcción de arquitecturas convolucionales ilustrado usando notación de expresiones regulares.	16
13	Aparición de algunas arquitecturas convolucionales notables con sus aportaciones.	16
14	Distribución de datos para la versión completa. En la parte superior de cada barra se muestra el porcentaje y número de imágenes para cada patología respecto al total (51,759)	21
15	Distribución de las etiquetas en las imágenes en la versión completa, considerando únicamente las que tienen al menos una patología asociada en la versión completa.	21
16	Distribución del número de pacientes de acuerdo a su sexo en la versión completa. Los porcentaje por categoría se presentan considerando el total de pacientes (14,402).	22
17	Distribución del número de imágenes por paciente en la versión completa.	22
18	Diagrama de cuerdas de la versión completa.	23
19	Distribución de los pacientes de acuerdo a grupos de edad en la versión completa.	23
20	Distribución de datos para la versión reducida. En la parte superior de cada barra se muestra el porcentaje y número de imágenes para cada patología respecto al total (880).	24
21	Distribución del número de pacientes de acuerdo a su sexo en la versión reducida. Los porcentaje por categoría se presentan considerando el total de pacientes (726).	24
22	Distribución del número de etiquetas por imagen en la versión reducida.	25

23	Distribución del número de imágenes por paciente en la versión reducida.	25
24	Diagrama de cuerdas de la versión reducida.	26
25	Distribución de los pacientes a los que pertenecen las radiografías de acuerdo a grupos de edad en la versión reducida.	26
26	Atelectasia.	27
27	Nódulo.	27
28	Infiltración.	28
29	Masa.	28
30	Derrame pleural.	29
31	Cardiomegalia.	29
32	Neumonía.	30
33	Neumotórax.	30
34	Diagrama general de procesamiento para clasificación multietiqueta de patologías.	34
35	Curvas ROC para distintos clasificadores.	35
36	Diagrama de experimentación para estudiar el impacto del aumentado de datos sobre la versión reducida de los datos.	37
37	Diagrama de experimentación para estudiar el impacto de la arquitectura sobre la versión reducida de los datos.	39
38	Diagrama de experimentación para estudiar el impacto de la transferencia de conocimiento sobre la versión reducida de los datos.	41
39	Diagrama de experimentación para estudiar el impacto del tamaño del conjunto de datos en el modelo de clasificación	43
40	Mejor configuración del modelo de clasificación de patologías.	44
41	Predicciones usando la configuración del modelo con mejor desempeño en la métrica uAP.	45
42	Predicciones usando la configuración del modelo con mejor desempeño en la métrica uAP.	46

Índice de cuadros

1	Detalles de algunas de las arquitecturas convolucionales más notables.	17
2	Comparativa de la cantidad de ejemplos, número de clases y el número de etiquetas entre la versión reducida y completa. Cada patología tiene asignada una etiqueta binaria.	20
3	Número de ejemplos en los subconjuntos generados mediante estratificación iterativa para la versión completa y reducida.	31
4	Número de ejemplos por clase para los subconjuntos de la versión completa. . .	32
5	Número de ejemplos por clase para los subconjuntos de la versión reducida. . .	32
6	Recursos de hardware utilizados en el proyecto.	36
7	Resultados para el modelo usando aumentado de datos.	38
8	Resultados en porcentaje (%) del AUC-ROC por patología usando aumentado de datos.	39

9	Resultados para el modelo usando diferentes arquitecturas convolucionales. . .	40
10	Resultados en porcentaje (%) del AUC-ROC por patología usando diferentes arquitecturas convolucionales.	40
11	Resultados para el modelo usando transferencia de conocimiento.	42
12	Resultados en porcentaje (%) del AUC-ROC por patología con transferencia de conocimiento.	42
13	Resultados del modelo con la mejor configuración para la versión completa y reducida.	44
14	Resultados en porcentaje (%) del conjunto de prueba en ambas versiones del conjunto de datos.	44

1 Introducción

Los algoritmos de inteligencia artificial han presentado un gran avance en las tareas relacionadas al reconocimiento de imágenes, siendo capaces de identificar patrones complejos. La clasificación de patologías a partir de radiografías consiste en identificar patrones visuales comúnmente asociados a cada una de las categorías. La complejidad de la tarea puede darse en distintos niveles. En el primer nivel, consiste en identificar sólo una patología por imagen. En un segundo nivel, se busca identificar múltiples patologías por imagen. El tipo de clasificación que se aborda en este proyecto considera casos donde se encuentra más de una patología por imagen. Las radiografías del cuerpo humano permiten al especialista hacer mejores evaluaciones de la condición médica de un paciente en diversos casos. Sin embargo, como una tarea especializada, el análisis de radiografías requiere de un alto nivel de conocimiento teórico y técnico para llevarse a cabo de manera eficaz. La identificación de objetos comunes, como frutas, animales, automóviles o casas, es una tarea sencilla. Por otro lado, las patologías presentes en una radiografía no son fáciles de distinguir y mucho menos de identificar de forma específica. Dado lo anterior, es importante el desarrollo de sistemas computacionales para apoyar a los especialistas en esta rigurosa labor, donde incluso médicos expertos discrepan en el diagnóstico de algunos casos.

Este proyecto realiza un análisis de desempeño de técnicas de aprendizaje profundo para la tarea de clasificación de patologías en radiografías de tórax. Con esto se busca contribuir en el desarrollo de sistemas de apoyo al especialista en el diagnóstico médico de este tipo de afecciones.

1.1 Contexto

La medicina es una disciplina enfocada en procurar la salud de las personas a través del estudio, prevención y tratamiento de enfermedades. Las imágenes médicas juegan un papel crucial en el diagnóstico médico de pacientes. Contribuyen a dar mejores y más precisos diagnósticos al proveer de un registro visual del estado interno de los órganos del cuerpo humano sin necesidad de hacer incisiones al paciente. En las últimas décadas se han producido importantes avances tecnológicos en el proceso de obtención de las diversas modalidades de imágenes médicas [1].

Una de las modalidades de imágenes más utilizadas son las radiografías. El proceso de obtención de estas imágenes consiste en exponer el cuerpo del paciente a un haz de radiación electromagnética, como resultado se generan imágenes en escala de grises de tejidos y estructuras corporales. Específicamente, las radiografías de tórax son un método de detección primario para enfermedades respiratorias y pulmonares. Este tipo de estudios son realizados a pacientes que presentan síntomas específicos y que pueden estar relacionados con patologías respiratorias. En algunas ocasiones también se realizan este tipo de pruebas como parte de evaluaciones médicas de rutina.

La interpretación de radiografías requiere de un conocimiento especializado. En particular, la interpretación de las radiografías de tórax es una tarea compleja debido a que existe una gran cantidad de estructuras expuestas en las imágenes, superposición de tejidos, líquidos, grasas y densidades óseas [1]. Además, los patrones visuales asociados a las patologías son muy similares, lo que dificulta aún más la interpretación de las imágenes.

En el aspecto computacional, los métodos del estado del arte para análisis de imágenes se basan en redes neuronales artificiales (RNA). Este tipo de métodos se han usado para clasificación [2], detección [3], segmentación [4] y generación de imágenes [5], etc. En medicina, las RNA se han aplicado para comenzar a resolver algunas de estas tareas [6]-[12]. Estos trabajos son precursores en el desarrollo de nuevas soluciones con mejoras en diferentes aspectos. Uno de ellos es el desempeño y confiabilidad en la predicción de patologías. Además, se busca que se integren de manera sencilla y efectiva en el proceso de diagnóstico para agilizarlo.

1.2 Problemática

La interpretación de radiografías es una tarea que requiere de un alto grado de especialización. Su complejidad está directamente relacionada con la zona corporal de la que sea obtenida la imagen. Existen diversos aspectos que dificultan la interpretación de las radiografías de tórax. Uno de ellos es la gran similitud entre los patrones asociados a múltiples patologías. Otros factores que influyen en la identificación de patologías son la calidad de las imágenes, la postura del paciente y la calibración del equipo radiológico.

La apropiada interpretación de las radiografías de tórax es principalmente llevada a cabo por radiólogos. El alto nivel de especialización de estos profesionales hace que en general sean escasos. En particular, en zonas rurales la demanda de interpretación supera por mucho la disponibilidad de estos expertos.

Es por estos motivos que contar con sistemas computacionales que apoyen al diagnóstico médico es de gran valor. Es importante enfatizar que estos sistemas están orientados únicamente a ser un recurso de ayuda. El diagnóstico final debe ser emitido por el radiólogo debido a la certificación de sus habilidades y su capacidad de tomar en cuenta otras fuentes de información como los reportes médicos.

1.3 Motivación

La medicina es una de las áreas del conocimiento más antiguas, pero al mismo tiempo una de las más presentes en la vida diaria y por ende con un gran impacto social. Brindar diagnósticos médicos precisos y oportunos es parte primordial de una atención médica de calidad. En este sentido, los avances tecnológicos como los sistemas computacionales han contribuido en el

apoyo al especialista, mejorando los diagnósticos y consigo los tratamientos, reduciendo errores humanos o respaldando decisiones del profesional para salvar vidas o evitar intervenciones innecesarias.

1.4 Objetivos

Desarrollar modelos de clasificación multietiqueta basados en redes neuronales artificiales para la clasificación de patologías asociadas a radiografías de tórax con base en evidencia visual de patrones anormales.

- Revisión del estado del arte de métodos basados en redes neuronales artificiales para la clasificación de objetos.
- Realizar un análisis exploratorio y cuantitativo de los conjuntos de datos de radiografías de tórax disponibles públicamente.
- Aplicar estrategias de preprocesamiento de datos en el conjunto de datos de radiografías de tórax a utilizar.
- Entrenar y ajustar métodos para la clasificación automática de patologías en radiografías de tórax.
- Realizar una evaluación de desempeño de los métodos entrenados.

2 Marco teórico

La inteligencia artificial (IA) es un campo de estudio, en el que intervienen las matemáticas y la ingeniería, que busca imitar algunas de las habilidades de los seres humanos como los sistemas de visión y reconocimiento de voz [13].

La inteligencia artificial puede dividirse en dos grandes tipos, la débil y la fuerte. La inteligencia artificial débil esta enfocada en resolver tareas específicas. Algunas aplicaciones de este tipo de inteligencia artificial son el reconocimiento de rostros, los asistentes de voz y los sistemas de recomendación. Por otro lado, la inteligencia artificial fuerte busca imitar la inteligencia y los comportamientos humanos por completo. Este tipo de IA pretende pensar, comprender, aprender y aplicar su inteligencia para resolver problemas complejos, al igual que los humanos. Actualmente, se han tenido avances importantes en resolver problemas únicamente dentro de la inteligencia artificial débil [14].

El aprendizaje automático es una rama de la inteligencia artificial que se enfoca en construir sistemas que aprenden de los datos. Este tipo de sistemas no requieren ser programados con reglas explícitas de comportamiento para desarrollar una tarea.

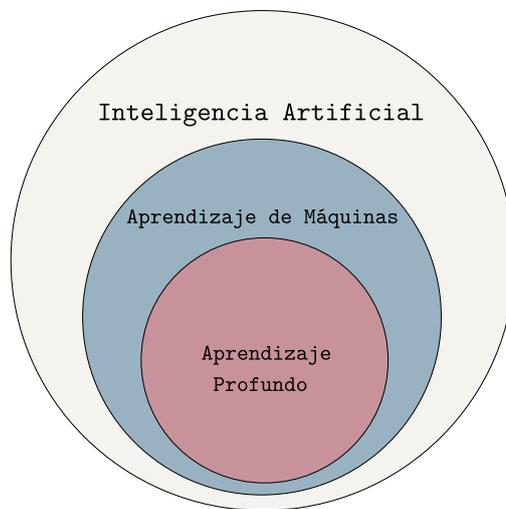


Figura 1: Relación entre inteligencia artificial, aprendizaje de máquinas y aprendizaje profundo.

El aprendizaje profundo es una rama dentro del aprendizaje de máquinas. Este campo se enfoca en aprender representaciones complejas de grandes conjuntos de datos usando redes neuronales artificiales. Las RNA son modelos computacionales que están levemente inspirados en el funcionamiento de las redes neuronales de los seres vivos [15]. Actualmente, las RNA son el estado del arte para diversas tareas en procesamiento de lenguaje natural [16]-[18], visión computacional [19]-[21], reconocimiento de voz y análisis de video [22], [23]. De manera gráfica, la

relación entre inteligencia artificial, aprendizaje de máquinas y aprendizaje profundo se ilustra en la Figura 1.

2.1 Tipos de aprendizaje

Los algoritmos de aprendizaje de máquinas se pueden categorizar de acuerdo a la manera que aprenden. La Figura 2 muestra un plano cartesiano que describe a grandes rasgos los tipos de aprendizaje y sus características. El eje horizontal en el mapa determina si un tipo de aprendizaje cuenta con instrucción o no, es decir, si el conjunto de datos cuenta con etiquetas o no. El eje vertical describe si el aprendizaje es activo o pasivo, es decir, si el algoritmo se retroalimenta del ambiente o no. En el primer cuadrante se tiene el aprendizaje supervisado, que se caracteriza por contar con conjuntos de datos etiquetados y no recibir retroalimentación del ambiente. Las principales tareas dentro de este tipo de aprendizaje son la clasificación y la regresión [24].

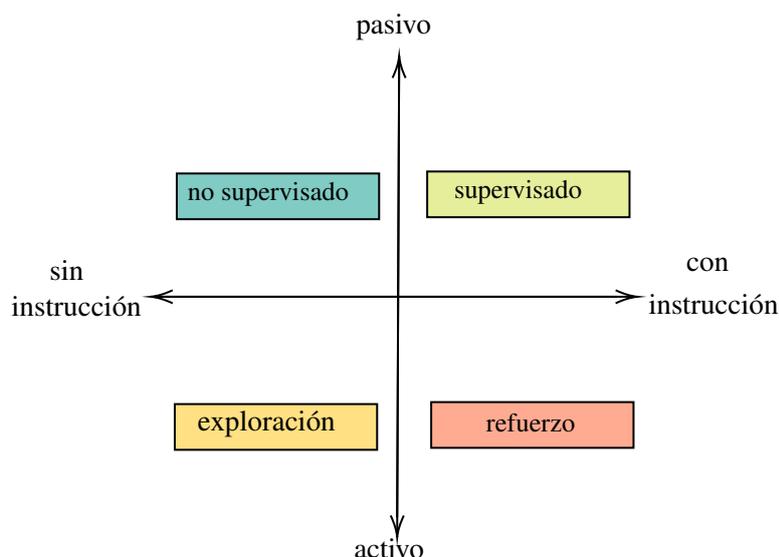


Figura 2: Tipos de aprendizaje en aprendizaje de máquinas.

El aprendizaje no supervisado se caracteriza por ser un aprendizaje pasivo y además, el conjunto de datos que usa no se encuentra etiquetado. Una de sus principales tareas es el agrupamiento.

El aprendizaje por refuerzo es un tipo de aprendizaje con instructor y activo. Su funcionamiento se enfoca en un proceso basado en la retroalimentación, el agente explora su entorno y con base en experiencias va mejorando su rendimiento. El agente es recompensado por cada buena acción y castigado por cada mala acción; por lo tanto, el objetivo del agente es maximizar las recompensas. Una de las aplicaciones de este tipo de aprendizaje son los videojuegos.

Por último, en el aprendizaje por exploración no se tiene información extra además de los datos y son agentes que buscan aprender de su entorno.

2.2 Tareas de clasificación en imágenes

Una de las principales tareas dentro del aprendizaje supervisado es la clasificación. Cuando se trabaja con imágenes, la clasificación consiste en asociar una imagen a una o múltiples etiquetas [25].

Los problemas de clasificación se pueden categorizar de acuerdo a la salida que se desea predecir. La clasificación binaria consiste en clasificar los datos de entrada en dos categorías mutuamente excluyentes. Por ejemplo, reconocer si en una imagen está una persona o no. La clasificación multiclase tiene como propósito asignar una etiqueta a la entrada de entre tres o más posibles. Por otro lado, la clasificación multietiqueta es una tarea de clasificación donde cada imagen puede estar asociada a más de una etiqueta, y algunas imágenes pueden contener todas las etiquetas posibles simultáneamente. Este tipo de tarea de clasificación es más compleja en comparación a las anteriores. En el dominio médico, la clasificación multietiqueta es una tarea recurrente debido a que las imágenes de un paciente pueden estar asociadas a más de una enfermedad. La Figura 3 ilustra las diferentes tareas de clasificación.

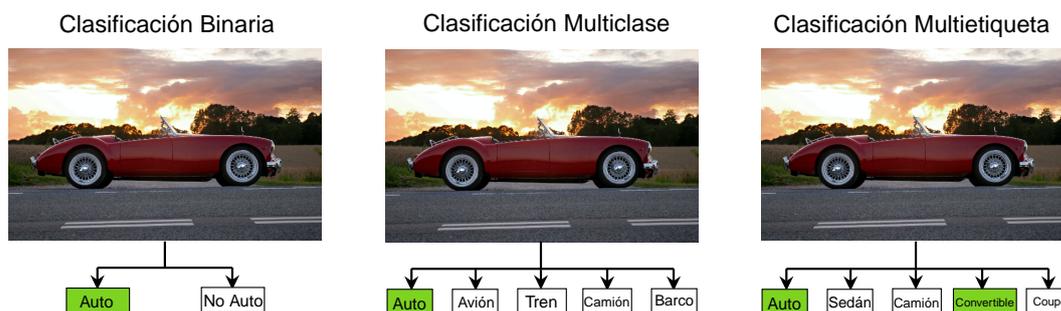


Figura 3: Problemas de clasificación.

En este proyecto se busca resolver un problema de clasificación de imágenes, lo que nos sitúa dentro del aprendizaje supervisado. Para resolver la tarea de clasificación se debe contar un conjunto de imágenes etiquetadas asociadas a la tarea que deseamos resolver. Se debe seleccionar un método parametrizado de aprendizaje, por ejemplo, una red neuronal. También, se debe seleccionar un algoritmo de optimización que actualice los parámetros del método a lo largo del entrenamiento. Finalmente, se debe elegir una función de pérdida, que es la encargada de guiar el entrenamiento. La idea de esta función es penalizar los parámetros del método si las categorías que predice el modelo son incorrectas.

El ciclo del aprendizaje para una tarea de clasificación se ilustra en la Figura 4 y funciona de la siguiente manera [26]:

1. Alimentar al método de aprendizaje con las imágenes de entrenamiento.

2. Realizar una predicción con los datos de entrada.
3. Calcular la pérdida usando las predicciones del método con las etiquetas en el conjunto.
4. Proponer nuevos parámetros para el método. El algoritmo de optimización toma la pérdida y los parámetros actuales y hace una nueva propuesta.

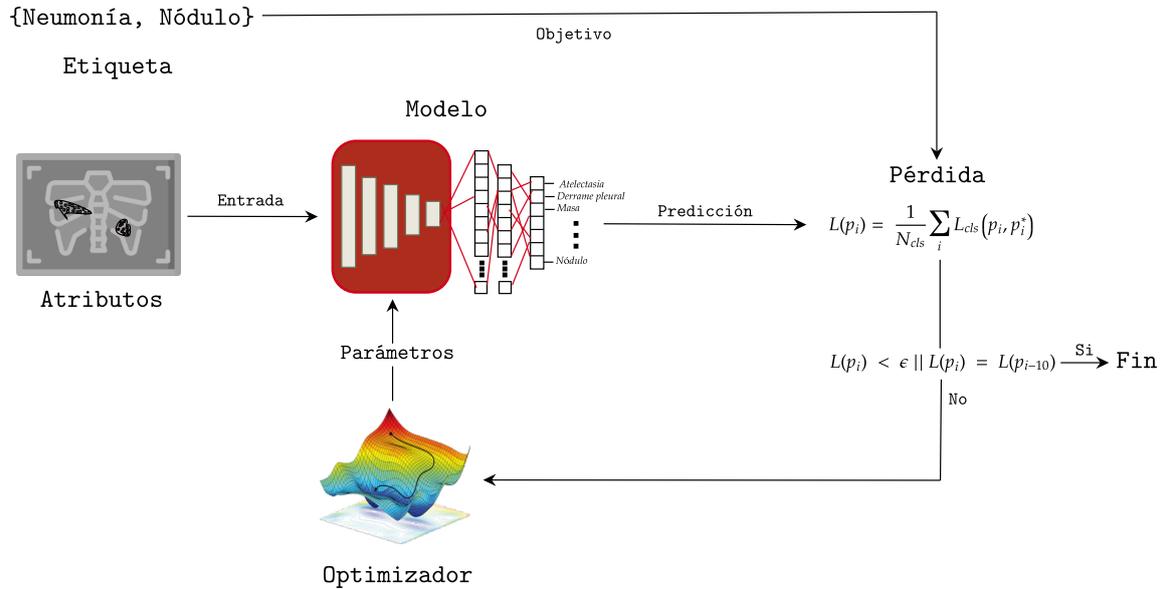


Figura 4: Ciclo de aprendizaje supervisado.

2.3 Redes neuronales artificiales

Las redes neuronales artificiales son modelos computacionales inspirados levemente en las redes neuronales biológicas que constituyen los cerebros de los seres vivos [27]. Se componen de unidades básicas de procesamiento llamadas neuronas artificiales que están conectadas entre sí en una red de comunicación que permite realizar cálculos de gran complejidad.

Actualmente, las RNA son usadas en varias áreas dentro de la inteligencia artificial. Algunas de ellas son visión computacional [19]-[21], procesamiento de lenguaje natural [16]-[18], reconocimiento de voz [28]-[30], etc.

2.3.1. Neurona artificial

La neurona artificial es la unidad de procesamiento más elemental dentro de una red neuronal [13], [31]. Toma un vector de entradas \vec{x} que se encuentra asociado a un vector de pesos \vec{w} , se realiza el producto punto entre ellos y se le suma un escalar conocido como sesgo b . Al resultado se le aplica una función no lineal conocida como función de activación σ . La Ecuación 1, mostrada

en [32], describe el procesamiento de la neurona artificial.

$$\hat{y} = \sigma\left(b + \sum_{i=1}^n w_i x_i\right) = \sigma(b + w^T \cdot x) \quad (1)$$

La Figura 5 muestra el diagrama de una neurona artificial.

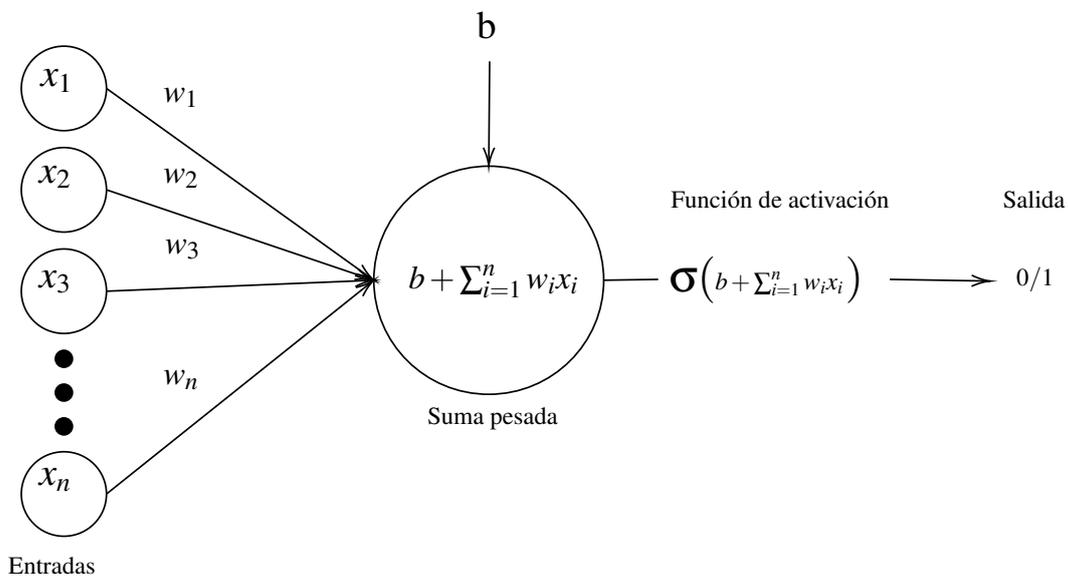


Figura 5: Neurona artificial.

Las entradas son las características o descriptores de un ejemplo en los datos sobre los que se desea aprender. Los pesos y el sesgo son parámetros entrenables que se buscan ajustar a valores que produzcan la salida deseada. Por último, la función de activación permite a la neurona aproximar soluciones a problemas no lineales.

2.3.2. Perceptrón multicapa

El perceptrón multicapa o red de capas completamente conectadas [33] es una red neuronal compuesta de grupos de neuronas organizadas en capas. Como se muestra en la Figura 6, las conexiones entre neuronas se dan entre capas de izquierda a derecha. La capa más a la izquierda es la capa de entrada, la capa más a la derecha es la capa de salida y las capas intermedias son las capas ocultas.

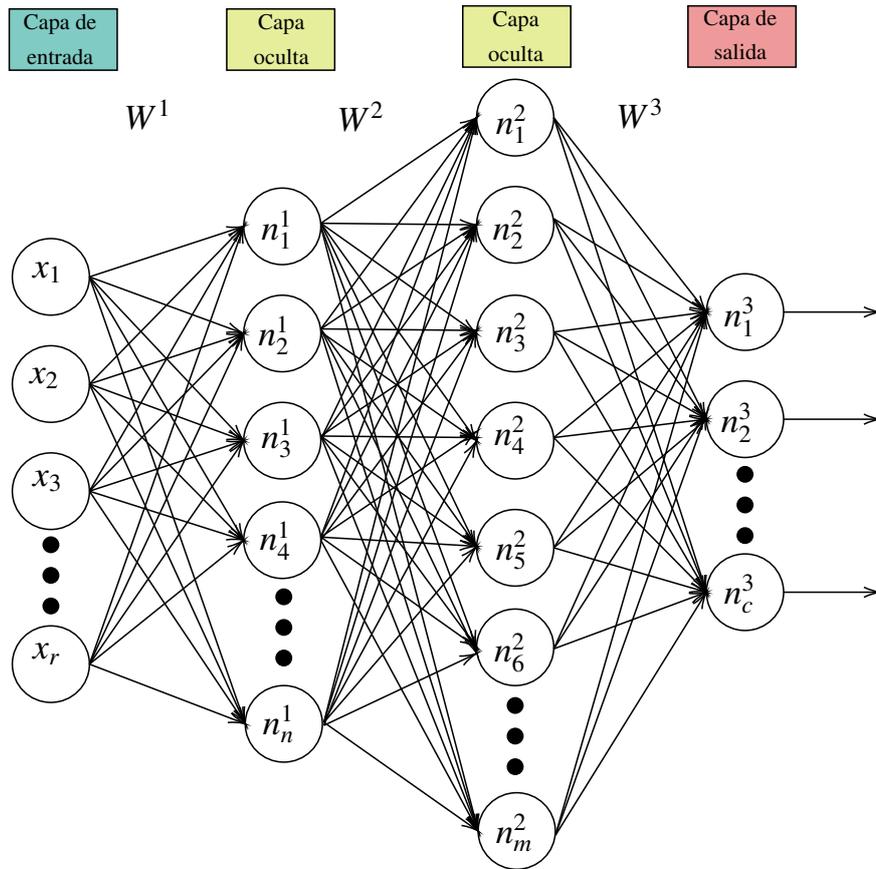


Figura 6: Perceptrón multicapa con dos capas ocultas.

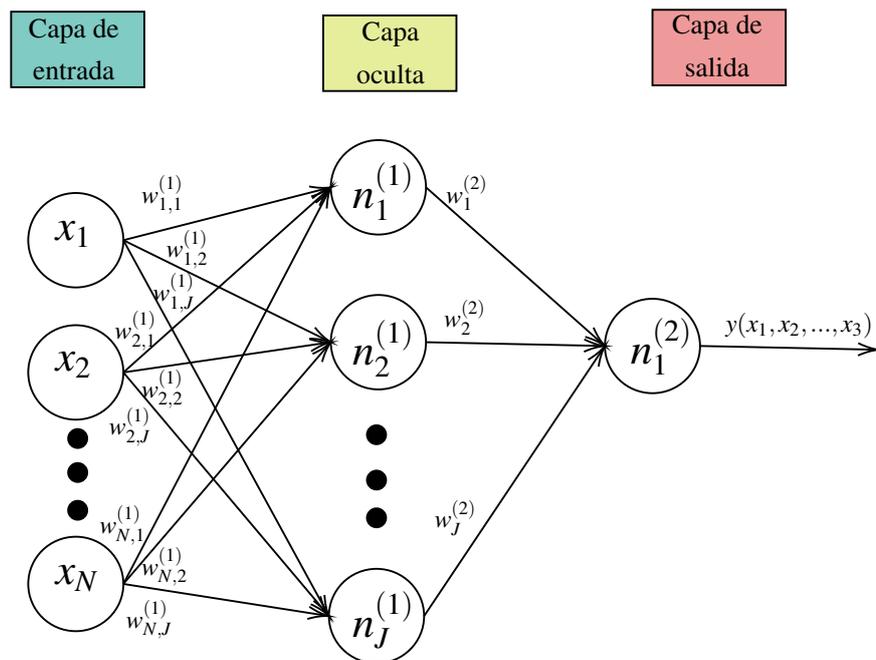


Figura 7: Perceptrón multicapa con una sola capa oculta y una neurona de salida.

El funcionamiento de un perceptrón multicapa es en dos pasos. En el primer paso conocido como paso hacia adelante, se realiza una propagación de la entrada de la red a cada una de las capas hasta obtener la salida. La Ecuación 2 describe el procesamiento que se da en el paso hacia adelante para el perceptrón de la Figura 7.

$$\hat{y}(x_1, x_2, \dots, x_N) = b^2 + \sum_{j=1}^J w_j^2 \sigma_j \left(b_j^1 + \sum_{n=1}^N w_{jn}^1 x_n \right) \quad (2)$$

Donde J es el número de neuronas en la capa oculta, σ la función de activación y w_j el peso entre la neurona j de la capa oculta y la neurona de la última capa. Además, w_{jn} es el peso entre el dato n de entrada y la neurona j de la capa oculta, y x_n los datos de la capa de entrada, siendo N el número total de elementos en dicha capa de entrada.

En el segundo paso conocido como paso hacia atrás, se emplea una función de pérdida. Esta función permite medir la diferencia entre el valor verdadero y las predicciones de la red neuronal. El objetivo en este punto es minimizar la pérdida de la red, para ello se emplea un algoritmo conocido como retro-propagación. Este algoritmo es iterativo y tiene como propósito ajustar los pesos y sesgos de la red de forma sistemática conforme a su influencia en el resultado final de la red. Para ello, se calcula el gradiente, es decir la derivada parcial del error obtenido de la función de pérdida respecto de los parámetros de la capa de salida de la red, con el cómputo de dicho gradiente este mismo es retropropagado hacia atrás para cada una de las neuronas en cada capa y con ello se ajustan los pesos y los sesgos. Los pasos hacia adelante y hacia atrás se realizan hasta que se cumpla un criterio de paro. Este criterio puede ser un número fijo de iteraciones o un cierto valor para el error.

La elección de la función de pérdida, así como la función de activación y el número de neuronas en la capa de salida, depende de la tarea a resolver.

2.3.3. Retropropagación

La propagación hacia atrás o retropropagación de errores es un algoritmo iterativo de optimización utilizado para entrenar redes neuronales artificiales. La idea general del algoritmo de retropropagación es minimizar una función de pérdida entre las etiquetas del conjunto de datos de entrenamiento y las predicciones del método [34].

Dada una red neuronal artificial y una función de error, el algoritmo calcula el gradiente de la función con respecto a los pesos de la red neuronal. El gradiente se calcula de adelante hacia atrás. Primero se calcula el gradiente de la capa final y se repite el proceso hasta llegar a la primera capa. Los cálculos parciales del gradiente de una capa se reutilizan en el cálculo del gradiente de la capa anterior. Este flujo hacia atrás de la información de error permite el cálculo eficiente del gradiente en cada capa en comparación con el enfoque ingenuo de calcular el gradiente de cada

capa por separado. Una vez que se tiene calculado el gradiente, es necesario usar dichos valores para actualizar los parámetros, para esto existen varios mecanismos de actualización [35]-[37].

2.4 Redes convolucionales

Una red neuronal convolucional (CNN) es un tipo de red neuronal artificial que se ha vuelto dominante en diversas tareas de análisis de imágenes [38]. La principal ventaja de las redes convolucionales para imágenes en comparación con métodos predecesores es que aprende características discriminantes directamente de los datos de forma automatizada. Por ejemplo, dadas muchas imágenes de gatos y perros, aprende las características discriminativas de cada clase por sí misma.

Las redes neuronales convolucionales tienen dos etapas de procesamiento, la extracción de características y la clasificación. La etapa de extracción de características se encarga de aprender representaciones espaciales jerárquicas de los datos de entrada. Esta etapa se compone de varias capas de convolución y submuestreo. La etapa de clasificación en las redes convolucionales es un perceptrón multicapa que discrimina las representaciones obtenidas en la etapa previa en las categorías de interés. En la Figura 8 se ilustra una arquitectura de red neuronal convolucional con dos capas de convolución y dos capas de submuestreo.

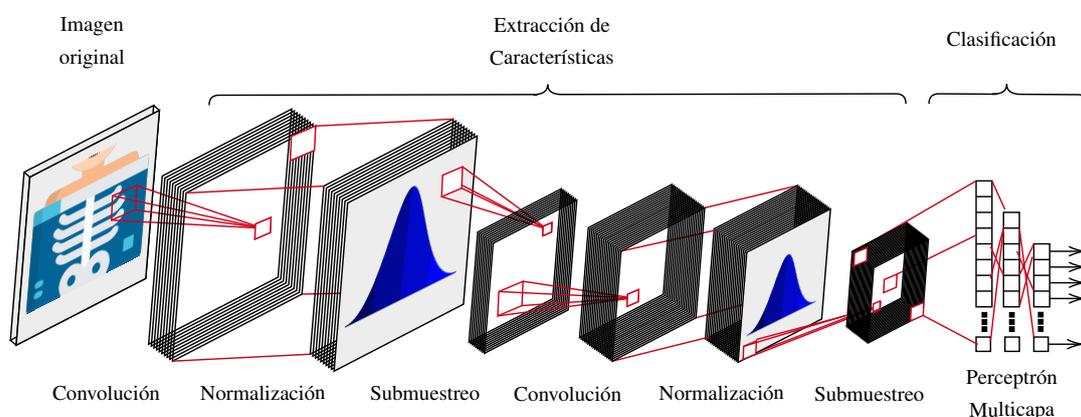


Figura 8: Etapas de procesamiento de una red neuronal convolucional. Primero, se toma como entrada una imagen y obtienen sus representaciones por la etapa de extracción de características. Después, en la etapa de clasificación, un perceptrón multicapa discrimina las representaciones en las categorías de interés.

2.4.1. Representación de imágenes

Una imagen digital en una computadora es representada por una matriz de píxeles de dimensiones $h \times w \times d$, donde h es el alto, w el ancho de la imagen y d es el número de canales. Cada píxel está representado por un valor numérico que indica su intensidad. En el caso de las imágenes en

escala de grises $d = 1$ y cada píxel tiene un valor entre 0 y 255, donde cero corresponde a negro y 255 a blanco. Los valores intermedios son distintos tonos de gris.

En el caso de las imágenes en espacio de color RGB el número de canales es 3 ($d = 3$). La representación es un volumen compuesto por tres matrices de píxeles. Este tipo de imágenes se pueden ver como tres diferentes, una imagen en escala roja, una imagen en escala verde y una imagen en escala azul. Las tres imágenes apiladas una encima de la otra.

2.4.2. Capas de convolución

Una capa convolucional es un bloque constructor principal de una red convolucional. Toma como entrada un tensor de dimensiones $h \times w \times d$ y realiza la operación de convolución con k filtros de dimensiones $r \times r \times d$, donde $r < h$ y $r < w$. En el contexto de las redes neuronales convolucionales, un filtro es un conjunto de pesos que se ajusta durante el entrenamiento.

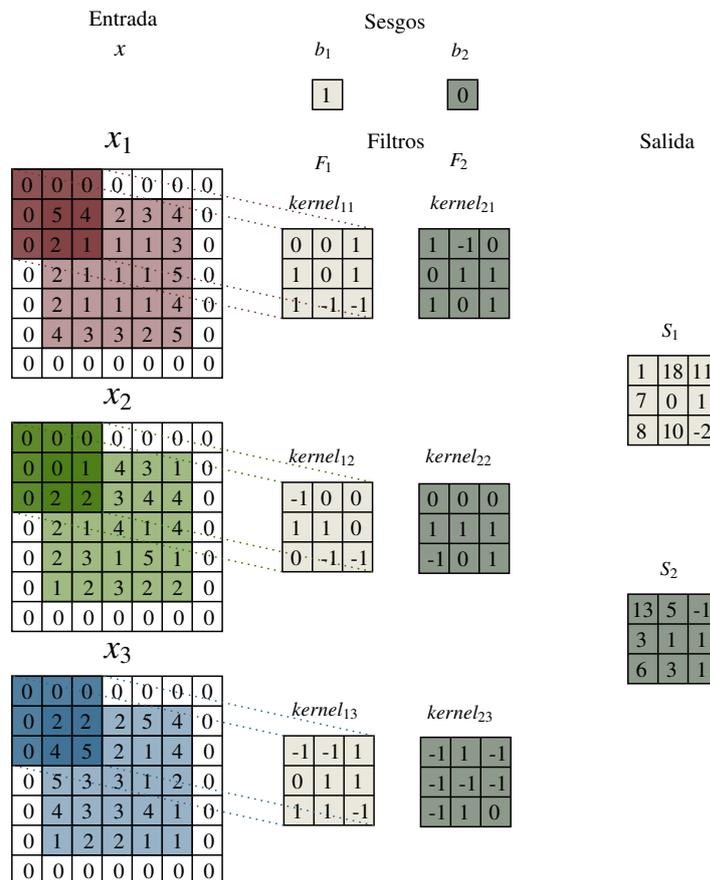


Figura 9: Procesamiento de una capa de convolución para una entrada de 5×5 con dos filtros de 3×3 , paso 2 y relleno de 1.

La convolución para un filtro se lleva a cabo de la siguiente manera. El filtro realiza un recorrido, a partir de la esquina superior izquierda de la matriz de entrada hasta la esquina inferior derecha.

El filtro se mueve de izquierda a derecha un número determinado de elementos y una vez que alcanza la esquina superior derecha, se mueve un elemento hacia abajo y se repite el proceso. En cada una de las posiciones que toma el filtro se calcula el producto elemento a elemento entre sus pesos y la entrada. Después, se realiza la suma de los valores y se le agrega un valor adicional que es el sesgo. Como resultado se obtiene una matriz de valores conocida como mapa de respuesta o mapa de características. Se realiza el mismo procedimiento para todos los filtros de la capa y por cada uno de ellos se obtiene un mapa de respuestas.

Los hiperparámetros en redes neuronales son parámetros ajustables por el diseñador que permiten controlar el proceso de entrenamiento. En el caso de las capas convolucionales se tienen tres hiperparámetros, el tamaño del filtro, el paso y el relleno. El paso determina cuantos elementos se desplazará el filtro. El tamaño del filtro indica las dimensiones espaciales, actualmente las arquitecturas del estado del arte usan filtros pequeños de 3×3 . El relleno consiste en agregar filas y columnas con el valor de 0 a la entrada de la capa de convolución para evitar la pérdida de información de los bordes al realizar la convolución [39].

La Figura 9 ilustra el procesamiento de una capa de convolución. La entrada es una matriz de dimensiones 5×5 con relleno de 1. La capa tiene dos filtros de 3×3 y el paso es de 2. Como salida se tienen dos mapas de respuesta de dimensiones 3×3 .

El tamaño de los mapas de respuesta está determinado por los valores de los hiperparámetros. Para realizar el cálculo de las dimensiones se tienen ecuaciones, para el alto del mapa se tiene la Ecuación 3 y para el ancho se tiene la Ecuación 4.

$$H^{(l)} = \frac{H^{(l-1)} + 2P^{(l-1)} - F_H^{(l)}}{S^{(l)}} + 1 \quad (3)$$

$$W^{(l)} = \frac{W^{(l-1)} + 2P^{(l-1)} - F_W^{(l)}}{S^{(l)}} + 1 \quad (4)$$

Donde $H^{(l)}$ y $W^{(l)}$ representan el alto y ancho respectivamente de la salida de la capa de convolución, mientras que $H^{(l-1)}$ y $W^{(l-1)}$ representan el alto y ancho del volumen de entrada. El término $P^{(l-1)}$ se conoce como relleno, este parámetro representa el número de píxeles que se agrega en los bordes del volumen de entrada. $F_H^{(l)}$ y $F_W^{(l)}$ son las dimensiones de alto y ancho de los filtros en la capa de convolución. El término $S^{(l)}$ se refiere al desplazamiento que realiza el filtro sobre el tensor de entrada.

2.4.3. Capas de submuestreo

Las capas de submuestreo se encargan de reducir las dimensiones espaciales de los mapas de características. Esto disminuye la cantidad de parámetros a aprender y la cantidad de cómputo

necesario para el entrenamiento de la red.

La capa de submuestreo reduce secciones del mapa de respuestas bajo algún criterio. Para ello, al igual que en la operación de convolución, se realiza un recorrido de izquierda a derecha y de arriba a abajo sobre el mapa de respuestas. Por cada grupo de valores de tamaño $r \times r$ se obtiene un valor, lo más frecuente es el promedio o el máximo. El proceso descrito se ilustra en la Figura 10.

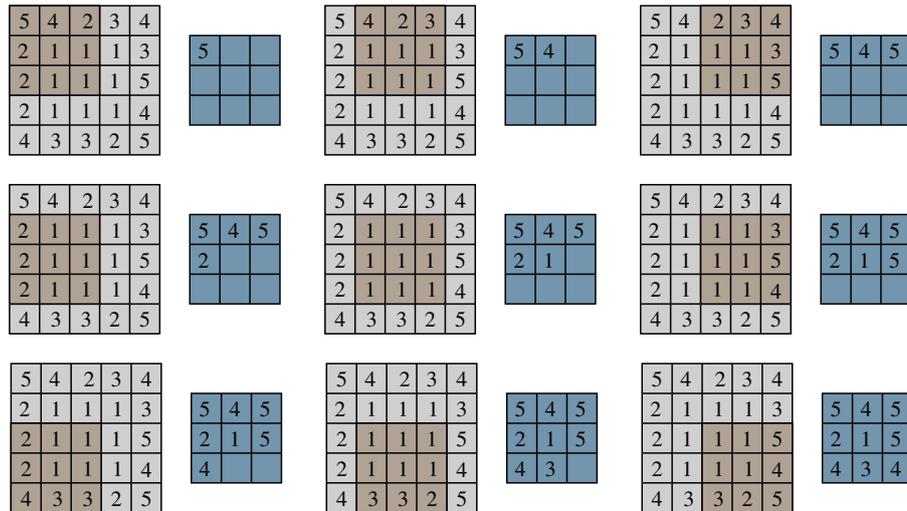


Figura 10: Procesamiento de una capa de submuestreo máximo para una entrada de 5×5 con paso de 1.

Un hiperparámetro de este tipo de capas es el salto que indica qué tan cercanas son las secciones de la entrada que se reducen.

2.4.4. Normalización

Las técnicas de normalización son esenciales para acelerar el entrenamiento y mejorar la generalización de las redes neuronales profundas [40], [41].

Existen diversas estrategias de normalización. Por ejemplo, la normalización por instancia se realiza por cada uno de los ejemplos que se propagan por la red. La normalización por grupo opera en la profundidad o canales del volumen de entrada, ya que los divide en grupos. Una de las estrategias más usadas en el entrenamiento de las RNA es la normalización por lotes [42]. Con esta estrategia de normalización, cada atributo de los ejemplos en el conjunto de datos se normaliza con una media cercana a cero y varianza unitaria, en función de sus estadísticas dentro del lote.

La normalización por lote con m ejemplos está dada por las expresiones que se encuentran en el bloque de Ecuaciones 5:

- Media del lote:

$$\mu_{\text{lote}} \leftarrow \frac{1}{m} \sum_{i=1}^m x^{(i)}$$

- Varianza del lote:

$$\sigma_{\text{lote}}^2 \leftarrow \frac{1}{m} \sum_{i=1}^m (x^{(i)} - \mu_{\text{lote}})^2 \quad (5)$$

- Normalización:

$$\hat{x}^{(i)} \leftarrow \frac{x^{(i)} - \mu_{\text{lote}}}{\sqrt{\sigma_{\text{lote}}^2 + \epsilon}}$$

- Escalado y desplazamiento:

$$y^{(i)} \leftarrow \gamma \hat{x}^{(i)} + \beta$$

En el cómputo de la normalización, se utiliza un término en el denominador para dar estabilidad numérica representado por ϵ , que es una constante arbitraria con valores pequeños cercanos a cero. Después de hacer la normalización con la media y la varianza calculada por lote, se hace un escalado de los datos del lote con una media β y una varianza γ . Los parámetros β y γ se inicializan como 0 y 1 respectivamente. Durante el entrenamiento se actualizan sus valores a una nueva media y varianza determinadas por el proceso de optimización. De manera gráfica, el funcionamiento de la normalización por lotes se ilustra en la Figura 11.

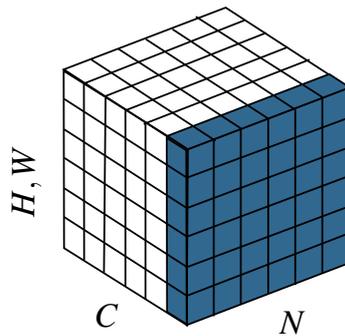


Figura 11: Dimensiones en las que opera la normalización por lote.

2.4.5. Arquitectura general

Si bien en la actualidad existen sofisticadas arquitecturas convolucionales [43]-[45], algunos patrones de construcción se basan en bloques que se repiten de formas distintas a lo largo de la red. La Figura 12 muestra un patrón básico para la construcción de arquitecturas de redes convolucionales.

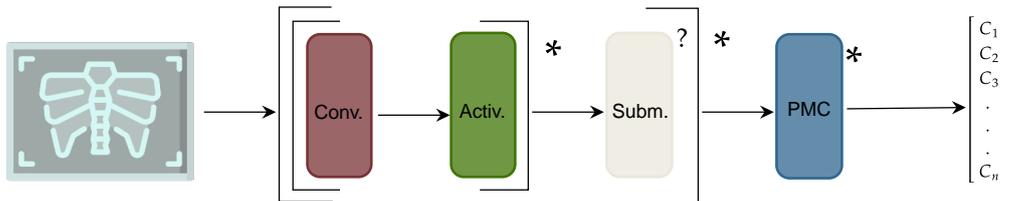


Figura 12: Patrón genérico de construcción de arquitecturas convolucionales ilustrado usando notación de expresiones regulares.

Este patrón consiste en la construcción de bloques básicos ([]) que pueden ser repetidos un cierto número de veces (*). Primero, se tiene un bloque constructor formado por una capa convulsional (Conv.) seguido de la función de activación (Activ.). Después de cada bloque convulsional puede seguir o no (?) una capa de submuestreo (Subm.). Finalmente, se tiene un perceptrón multicapa (PMC) para discriminar las características aprendidas.

Este patrón para la construcción de arquitecturas es muy útil, ya que las arquitecturas del estado del arte pueden requerir de muchos recursos de cómputo y mucho tiempo para su entrenamiento.

2.5 Arquitecturas notables

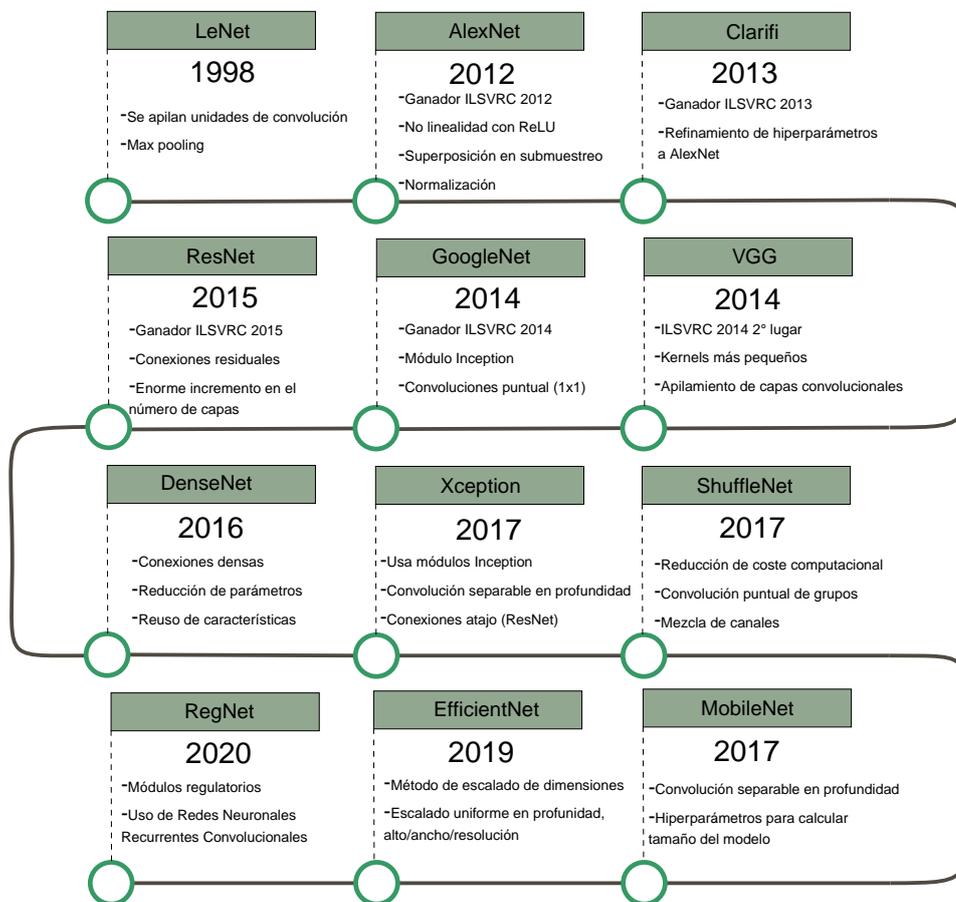


Figura 13: Aparición de algunas arquitecturas convolucionales notables con sus aportaciones.

A lo largo de los años, se han desarrollado variantes de las arquitecturas de redes convolucionales que han dado lugar a importantes avances en el campo de visión computacional [46]. Una buena medida de este progreso son las tasas de error que se han presentado en competencias como la ILSVRC ImageNet. El objetivo del desafío era promover el desarrollo de métodos para el análisis de imágenes. En esta competencia surgieron algunas de las arquitecturas convolucionales más notables de la historia. La Figura 13 muestra una cronología de la evolución de las arquitecturas convolucionales más sobresalientes con algunas de sus aportaciones [47].

De manera más detallada se describen en el Cuadro 1.

Datos importantes de las arquitecturas				
Arquitectura	Característica	Profundidad	Lanzamiento	Variaciones
LeNet	Tradicional	Conv(2) Pool(2) Dense(3)	1998	–
AlexNet	Desconexiones	Conv(5) Pool(3) Dense(3)	2012	–
Clarifai	Refinado de hiperparámetros	Conv(5) Pool(3) Dense(3)	2013	–
VGGNet	Incremento de profundidad	Conv(13) Pool(5) Dense(3)	2014	VGG(16,19)
GoogLeNet	Modulo Inception	Conv(21) Pool(5) Dense(1)	2014	Inception(V2,v3)
ResNet	Bloques Residual	Conv(152) Dense(1)	2015	ResNet(18,34,50,101,152)
DenseNet	Bloque Dense	Conv(5) Trans(84) Dense(32)	2016	DenseNet(121,169,201,264)
Xception	Módulo Stem	Conv(19) Pool(5) Dense(2)	2017	–
ShuffleNet	Convs DepthWise & Group	Stage2 Stage3 Stage4 Dense(1)	2017	ShuffleNet(v1, v2)
MobileNet	Convs DepthWise & PointWise	Conv(10) DWise(9) Dense(2)	2017	MobileNet(V2,V3)
EfficientNet	Operaciones Eficientes	Blocks(7)	2019	EfficientNet(b0-b7)
RegNet	Auto-regulación	Layers(20)	2020	RegNet(20,32,56)

Cuadro 1: Detalles de algunas de las arquitecturas convolucionales más notables.

2.6 Transferencia de conocimiento

La técnica de transferencia de conocimiento consiste en reutilizar el conocimiento aprendido por un modelo para resolver una tarea y aplicarlo a otra diferente pero relacionada. Por ejemplo, si se entrenó un modelo para clasificar perros y gatos, se puede reutilizar el conocimiento aprendido para reconocer otro tipo de mamíferos.

Para llevar a cabo la transferencia de conocimiento se entrena una arquitectura convolucional en un conjunto de datos a gran escala como ImageNet. Después, para resolver una nueva tarea, se toman la arquitectura y los pesos aprendidos y se ajustan para resolver la nueva tarea. La transferencia funciona debido a que las redes neuronales aprenden representaciones de manera jerárquica. Las primeras capas aprenden patrones sencillos como bordes y esquinas que son útiles para reconocer una gran cantidad de objetos, mientras que las capas más profundas aprenden representaciones cada vez más complejas [48].

La transferencia de conocimiento ha sido ampliamente usada en redes neuronales como una alternativa para reducir la cantidad de datos etiquetados requeridos para el entrenamiento [48], [49]. Esto es importante, ya que en ciertos dominios contar con datos etiquetados a gran escala no

es posible. Por ejemplo, en el ámbito médico es muy costoso debido al nivel de especialización necesaria para etiquetar los datos.

Actualmente, existen diversos recursos donde se pueden encontrar arquitecturas preentrenadas en conjuntos de datos muy grandes y generales que se pueden aplicar a diversas tareas ^{1,2}. Las arquitecturas convolucionales más sobresalientes como las mencionadas en el Cuadro 1 se encuentran disponibles, lo cual contribuye en la resolución de problemas.

¹<https://pytorch.org/vision/stable/models.html>

²<https://github.com/tensorflow/models/tree/master/official>

3 Conjunto de datos ChestX-ray8

ChestX-ray8 es un conjunto de datos multietiqueta de radiografías de tórax publicado por el Instituto Administrador de la Investigación Médica en Estados Unidos (NIH, por sus siglas en inglés) en 2017 [50]. El conjunto cuenta con 112,120 imágenes en formato PNG con una resolución de 1024×1024 píxeles. Las patologías a las cuales están asociadas las imágenes son: *atelectasia*, *cardiomegalia*, *derrame pleural*, *infiltración*, *masa*, *nódulo*, *neumonía* y *neumotórax*. Además, contiene una categoría de *sin hallazgos* que corresponde a las radiografías que no fueron asociadas a ninguna de las patologías antes mencionadas.

El etiquetado del conjunto de datos se realizó de manera automatizada con técnicas de procesamiento de lenguaje natural (PLN) [51]. A partir de los reportes radiológicos se identificaron palabras clave asociadas a las patologías de interés, lo que permitió el etiquetado de las imágenes. Posteriormente, se validó como parte del mismo trabajo presentado en 2017, el etiquetado de un subconjunto de 880 radiografías por un radiólogo.

Después de publicado *ChestX-ray8* se presentó una extensión al conjunto nombrado *ChestX-ray14*. En este último conjunto se incrementaron el número de imágenes y además, se extendieron a catorce patologías. Se agregaron las categorías de *consolidación*, *edema*, *enfisema*, *fibrosis*, *engrosamiento pleural* y *hernia*.

El conjunto de datos se encuentra disponible a través del repositorio ChestXray-NIHCC³. Los archivos contenidos en el repositorio son los siguientes:

- *train_val_list.txt* y *test_list.txt*: son archivos que en conjunto contienen el listado del total de imágenes. Los archivos se encuentran separados pues es la partición provista para el subconjunto de entrenamiento y subconjunto de prueba.
- *Data_Entry_2017_v2020.csv*: archivo con metadatos de todas las imágenes. El archivo contiene las siguientes propiedades de las imágenes: *índice de la imagen*, *patologías asociadas (etiquetas)*, *identificador único de cada paciente*, *edad paciente*, *sexo del paciente*, *posición del paciente (vista de la radiografía)*, *tamaño original de la imagen* y el *espaciado original entre píxeles*.
- *BBox_List_2017.csv*: contiene los metadatos de las 880 imágenes que tienen etiquetado manual. En este archivo se almacenan las siguientes propiedades: *índice de la imagen*, *patología asociada* y las coordenadas de la localización de la patología en la radiografía en formato $[X, Y, W, H]$.

³<https://nihcc.app.box.com/v/ChestXray-NIHCC/>

3.1 Motivación de uso

Para el desarrollo de este proyecto se emplearon tanto *ChestX-ray8* como *ChestX-ray14* por las siguientes razones. Primero, las imágenes y los metadatos del conjunto se encuentran disponibles, y cuenta con etiquetado multietiqueta. Segundo, el etiquetado de un subconjunto de imágenes fue validado por un experto radiólogo desde la construcción original del conjunto de datos. Además, este conjunto presenta gran variabilidad en las radiografías, lo que puede contribuir en la construcción de un modelo de clasificación robusto. Por ejemplo, las imágenes corresponden a pacientes dentro un amplio rango de edades y contiene radiografías de mujeres y hombres.

3.2 Versiones reducida y completa

En este proyecto, a partir de los conjuntos originales se derivaron dos versiones de conjuntos de datos que se denominaron como versión completa y versión reducida.

La versión completa toma como base *ChestX-ray14* y cuenta con 51,759 imágenes, donde cada imagen está asociada con al menos una patología. Esta versión, como se mencionó anteriormente, cuenta con un etiquetado que fue realizado de manera automática. Las imágenes en esta versión se encuentran asociadas a las 14 patologías.

La versión reducida toma como base *ChestX-ray8* y cuenta con 880 imágenes, que además de ser etiquetadas automáticamente fueron revisadas por el experto radiólogo. Estas imágenes también pertenecen a la versión completa. Sin embargo, este subconjunto sólo cuenta con las ocho patologías con las que fue presentado el conjunto inicialmente.

La diferencia principal entre las versiones es el tamaño. Esto toma relevancia debido a que las redes neuronales emplean muchos datos para su entrenamiento. Por lo tanto, ambas versiones del conjunto fueron usadas para entrenar modelos de clasificación y después se realizó una comparación en desempeño. El Cuadro 2 muestra las diferencias en la cantidad de ejemplos, número de patologías y número de etiquetas.

Versión	Número de ejemplos	Número de patologías	Etiquetas asignadas
Reducida	880	8	984
Completa	51,759	14	81,176

Cuadro 2: Comparativa de la cantidad de ejemplos, número de clases y el número de etiquetas entre la versión reducida y completa. Cada patología tiene asignada una etiqueta binaria.

3.3 Análisis exploratorio de los datos

En esta sección se presentan estadísticas importantes de ambas versiones del conjunto de datos.

3.3.1. Estadísticas de la versión completa

La Figura 14 muestra la distribución de las clases de acuerdo a las etiquetas.

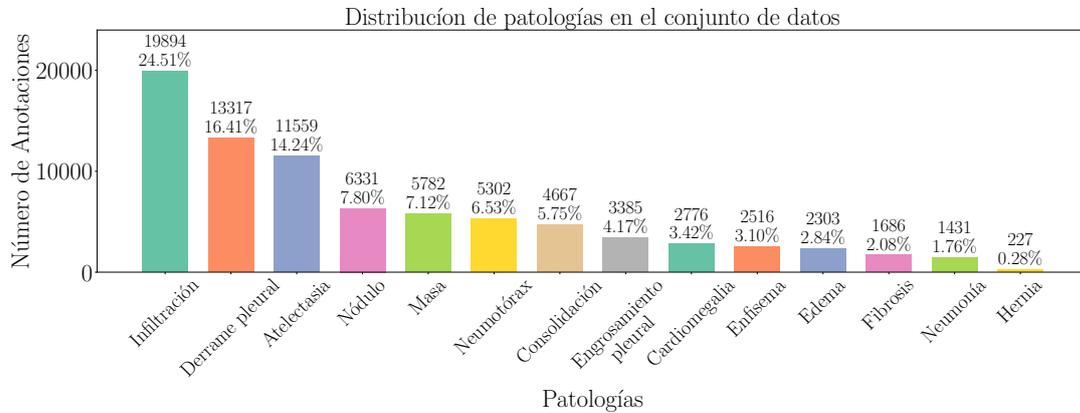


Figura 14: Distribución de datos para la versión completa. En la parte superior de cada barra se muestra el porcentaje y número de imágenes para cada patología respecto al total (51,759)

La Figura 15 ilustra la distribución del número de etiquetas en las imágenes, considerando únicamente las que tienen al menos una patología asociada.

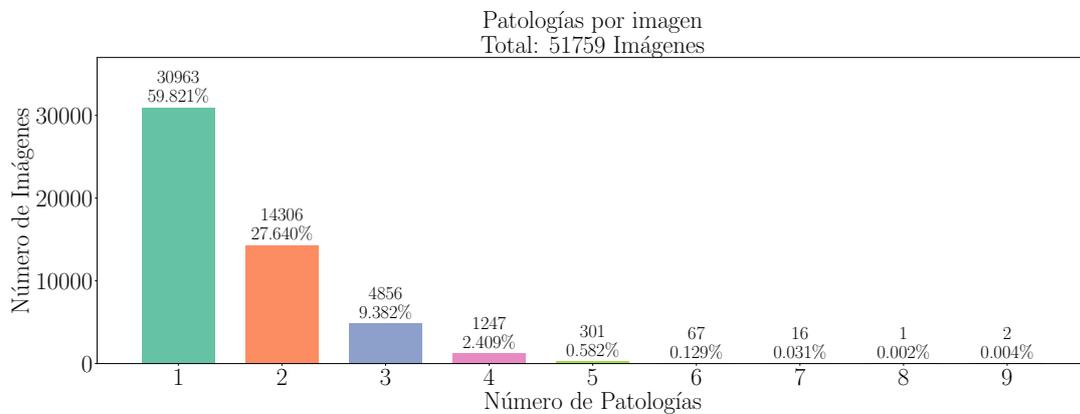


Figura 15: Distribución de las etiquetas en las imágenes en la versión completa, considerando únicamente las que tienen al menos una patología asociada en la versión completa.

Como se puede apreciar, existe una gran cantidad de imágenes en la categoría de *sin hallazgos*, dichos casos no fueron utilizados para mantener consistencia con las condiciones de ambas versiones del conjunto de datos. Por otro lado, existe un marcado desbalance en el número de ejemplos asociados a las patologías. Algunas estadísticas generales de la versión completa son:

- Número de radiografías: 51,759.
- Número de pacientes únicos: 14,402.
- Número de etiquetas asignadas: 81,176.

Dos de los aspectos más importantes de un conjunto de datos multietiqueta son la cardinalidad y la densidad de las etiquetas [52]. La cardinalidad se define como el número promedio de etiquetas asociadas a un ejemplo en el conjunto. La densidad se calcula como la cardinalidad dividida por el número de categorías presentes.

- Cardinalidad: 1.5683.
- Densidad: 0.1120.

El sexo del paciente es un aspecto que influye drásticamente en la apariencia de las radiografías de tórax debido a las diferencias anatómicas entre hombres y mujeres. La Figura 16 ilustra la distribución de los pacientes de acuerdo a su sexo.

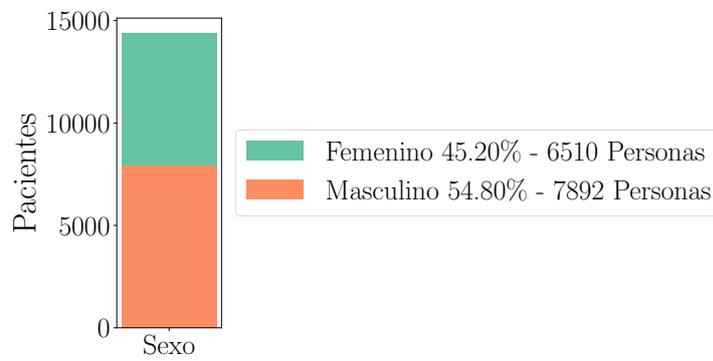


Figura 16: Distribución del número de pacientes de acuerdo a su sexo en la versión completa. Los porcentajes por categoría se presentan considerando el total de pacientes (14,402).

La Figura 17 muestra la distribución del número de imágenes por paciente.

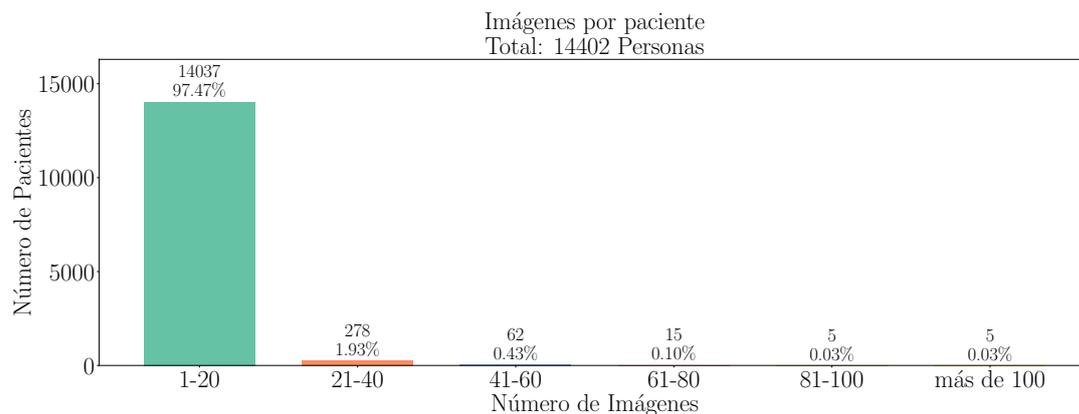


Figura 17: Distribución del número de imágenes por paciente en la versión completa.

La Figura 18 ilustra el diagrama de cuerdas asociado a la versión completa del conjunto de datos. Este diagrama permite visualizar las relaciones entre las patologías presentes en el conjunto. Las

cuerdas entre las patologías indican una relación entre estas, mientras que el grosor de la cuerda indica que tan relacionadas están las patologías. De esta figura se aprecia que existe una mayor relación entre *infiltración*, *derrame pleural* y *atelectasia*.

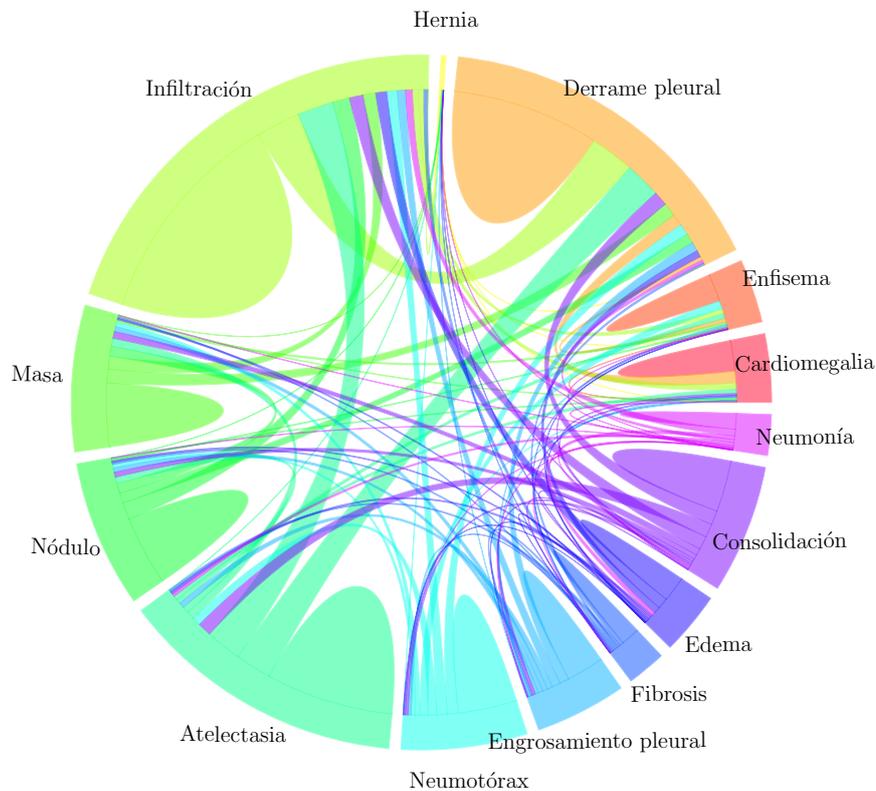
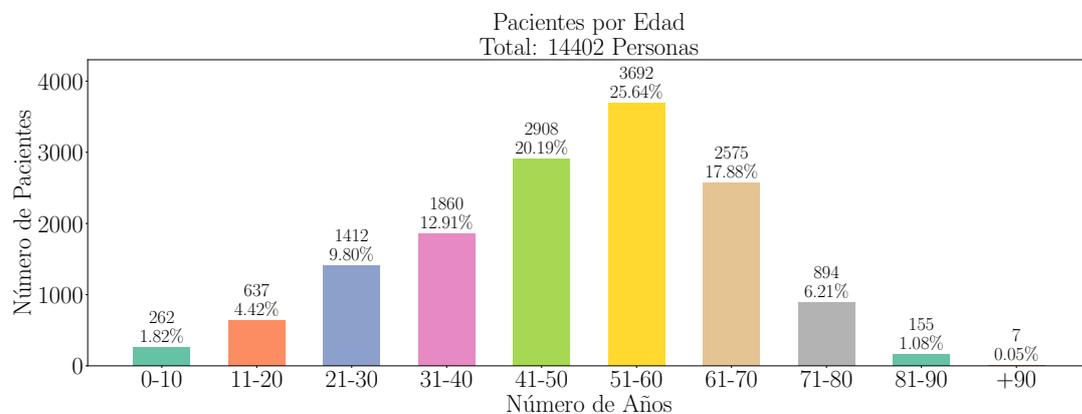


Figura 18: Diagrama de cuerdas de la versión completa.

La Figura 19 ilustra la distribución de los pacientes de acuerdo a diferentes grupos de edad. Se tienen mayor número de pacientes entre 51 y 60 años.



*

Figura 19: Distribución de los pacientes de acuerdo a grupos de edad en la versión completa.

3.3.2. Estadísticas de la versión reducida

La Figura 20 muestra la distribución de las patologías de acuerdo a las etiquetas para la versión reducida. Esta versión en comparación con la completa presenta un menor desbalance.

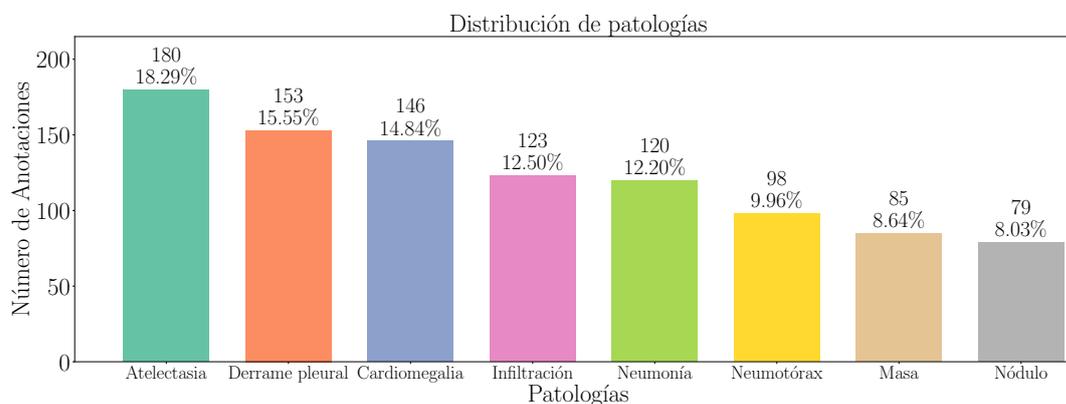


Figura 20: Distribución de datos para la versión reducida. En la parte superior de cada barra se muestra el porcentaje y número de imágenes para cada patología respecto al total (880).

Algunas estadísticas generales de la versión reducida son:

- Número de radiografías: 880.
- Número de pacientes únicos: 726.
- Número de etiquetas en el conjunto: 984.

La Figura 21 muestra la distribución de los pacientes a los que pertenecen las radiografías de acuerdo al sexo. Al igual que en la versión completa no se muestra un desbalance significativo.

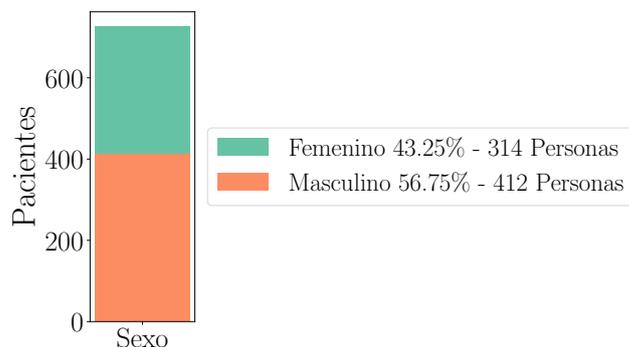


Figura 21: Distribución del número de pacientes de acuerdo a su sexo en la versión reducida. Los porcentaje por categoría se presentan considerando el total de pacientes (726).

La Figura 22 muestra la distribución del número de etiquetas por imagen. La mayoría de las imágenes poseen solo una etiqueta.

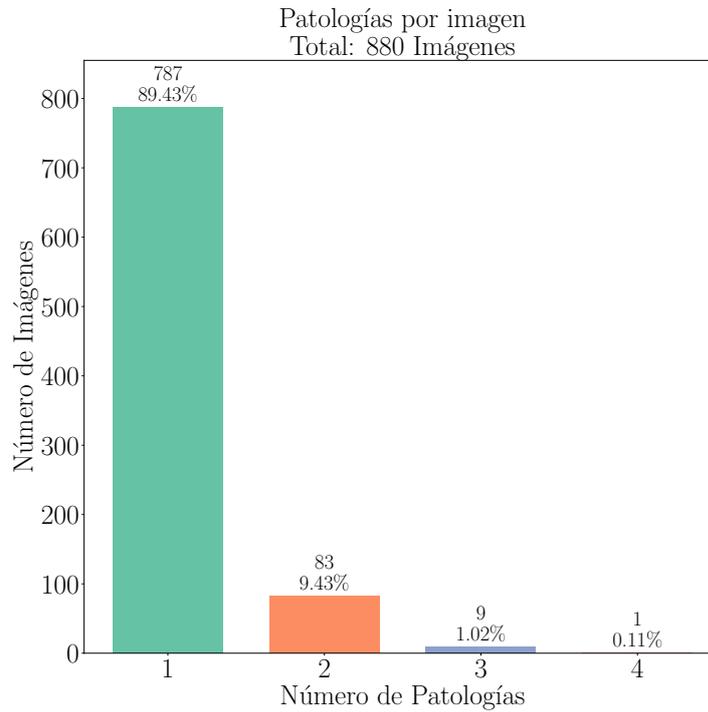


Figura 22: Distribución del número de etiquetas por imagen en la versión reducida.

La Figura 23 muestra la distribución del número de imágenes por paciente. La mayoría de pacientes (76.15 %) tiene una sola radiografía asociada.

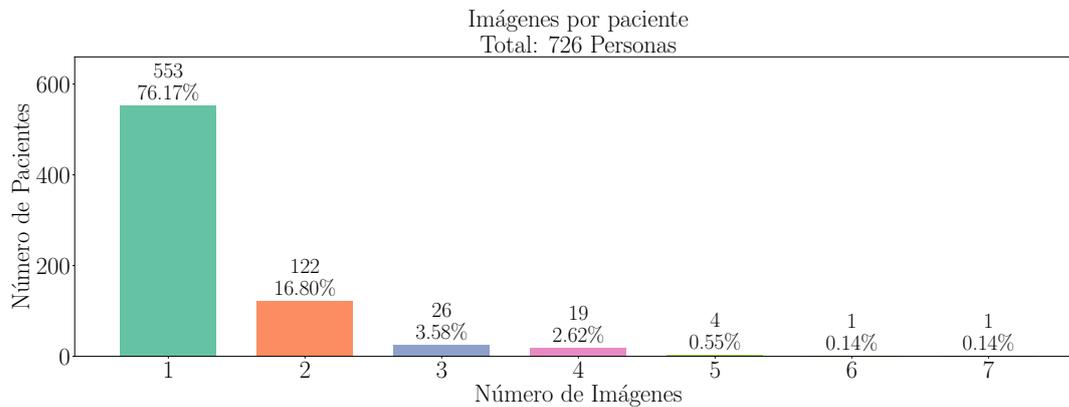


Figura 23: Distribución del número de imágenes por paciente en la versión reducida.

La cardinalidad y la densidad de las etiquetas son:

- Cardinalidad: 1.1182.
- Densidad: 0.1398.

La Figura 24 muestra el diagrama de cuerdas para la versión reducida. Se conservan las relaciones presentes en el diagrama de cuerdas de la versión completa.

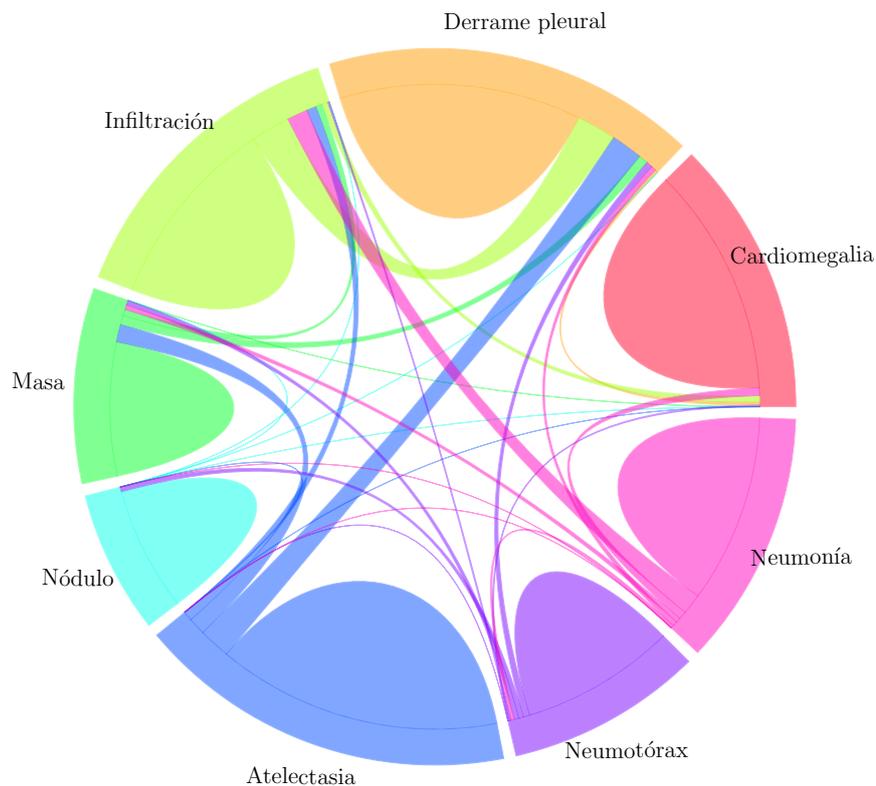


Figura 24: Diagrama de cuerdas de la versión reducida.

La Figura 25 ilustra la distribución de los pacientes a los que pertenecen las radiografías de acuerdo a diferentes grupos de edad. Al igual que en la versión completa, existe mayor número de pacientes con una edad entre 51 y 60 años.

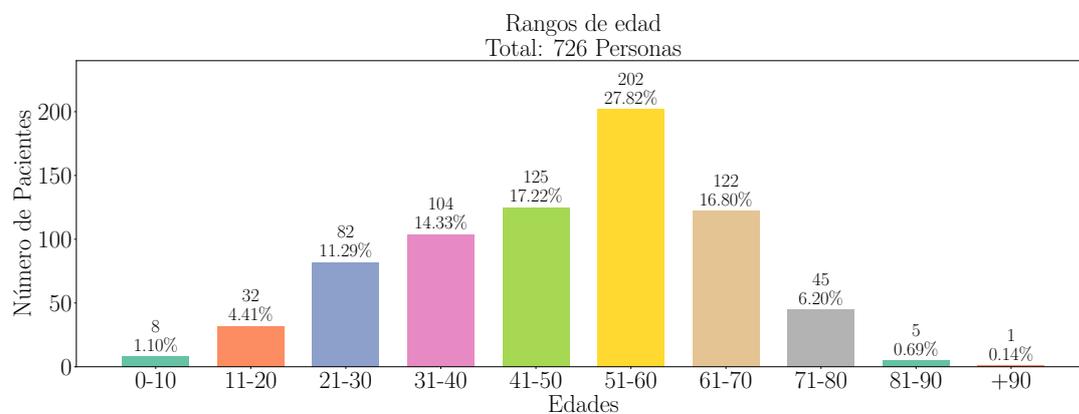


Figura 25: Distribución de los pacientes a los que pertenecen las radiografías de acuerdo a grupos de edad en la versión reducida.

3.3.3. Ejemplos del conjunto de datos

Se presentan imágenes pertenecientes a la versión reducida debido a que cuentan con etiquetado manual de las patologías presentes, y con datos sobre la ubicación de las mismas en cada imagen. En las figuras [26-33] se muestra una imagen por cada patología presente en la versión reducida.

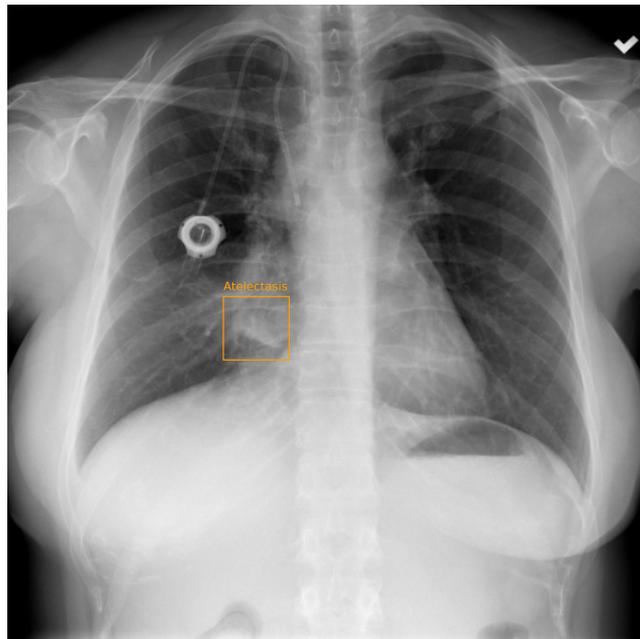


Figura 26: Atelectasia.

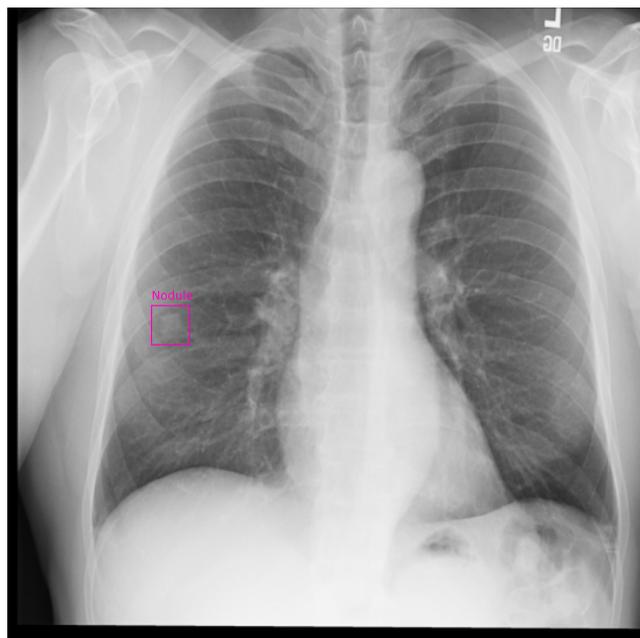


Figura 27: Nódulo.

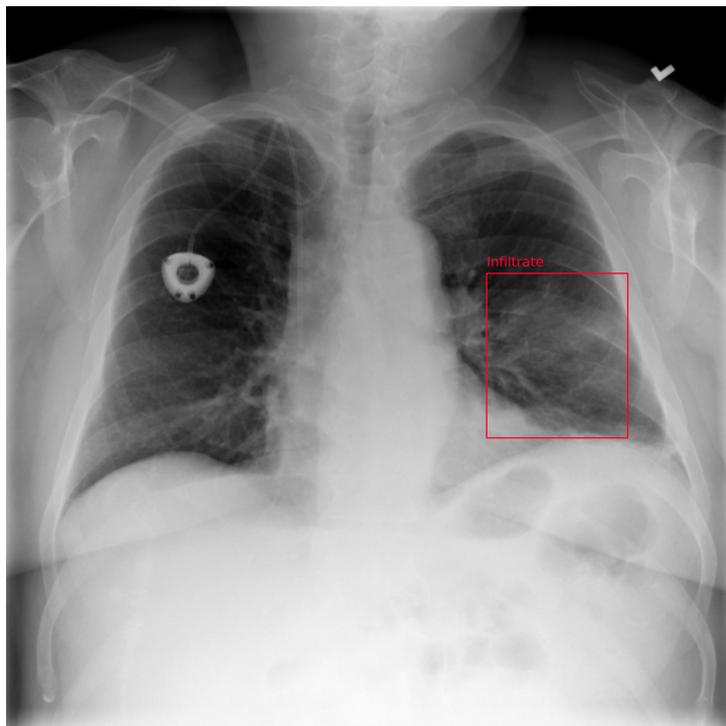


Figura 28: Infiltración.

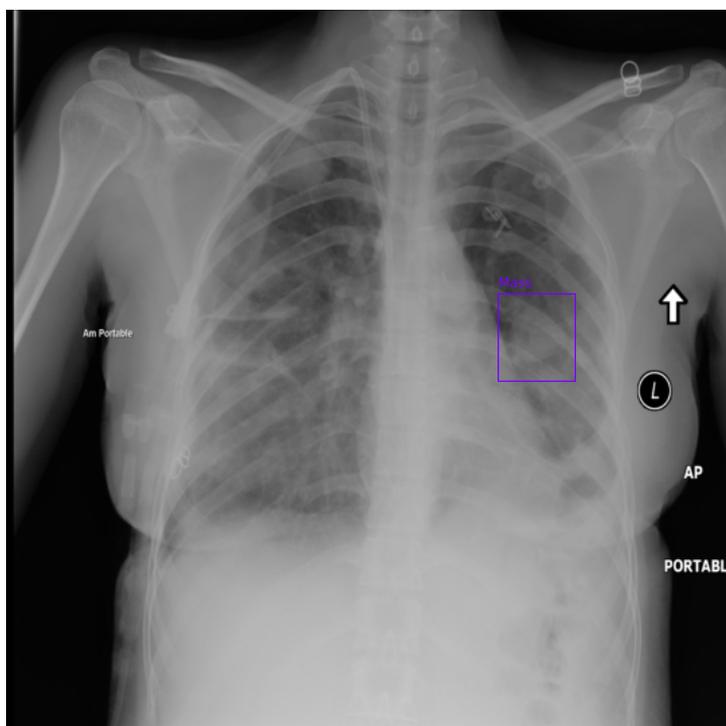


Figura 29: Masa.

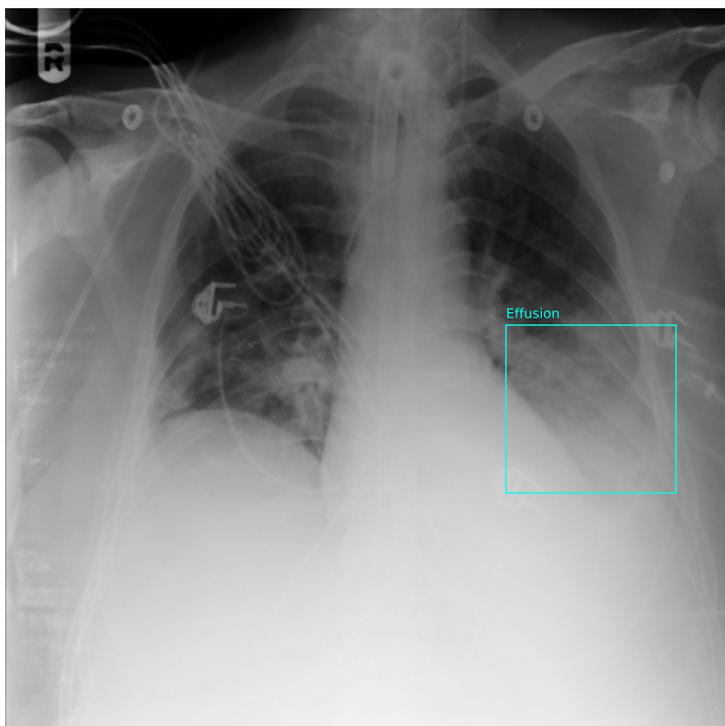


Figura 30: Derrame pleural.

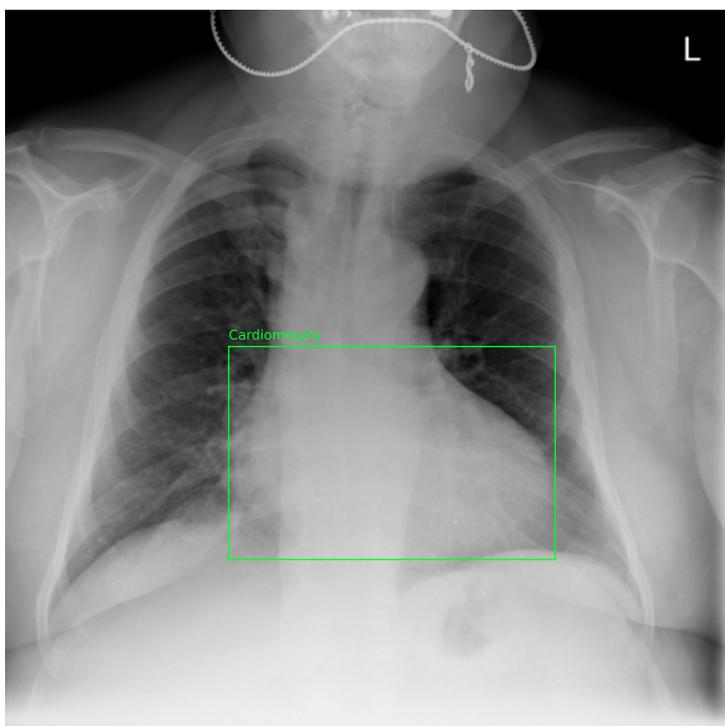


Figura 31: Cardiomegalia.

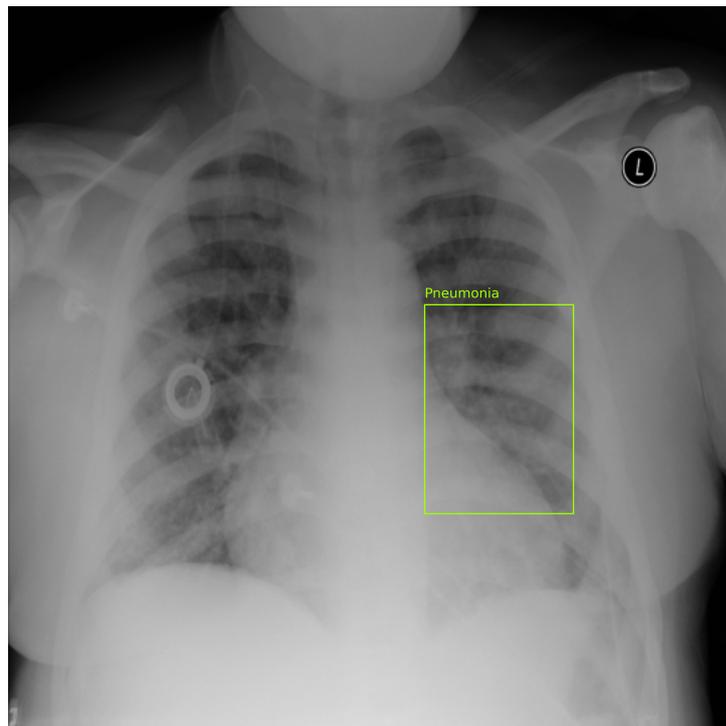


Figura 32: Neumonía.

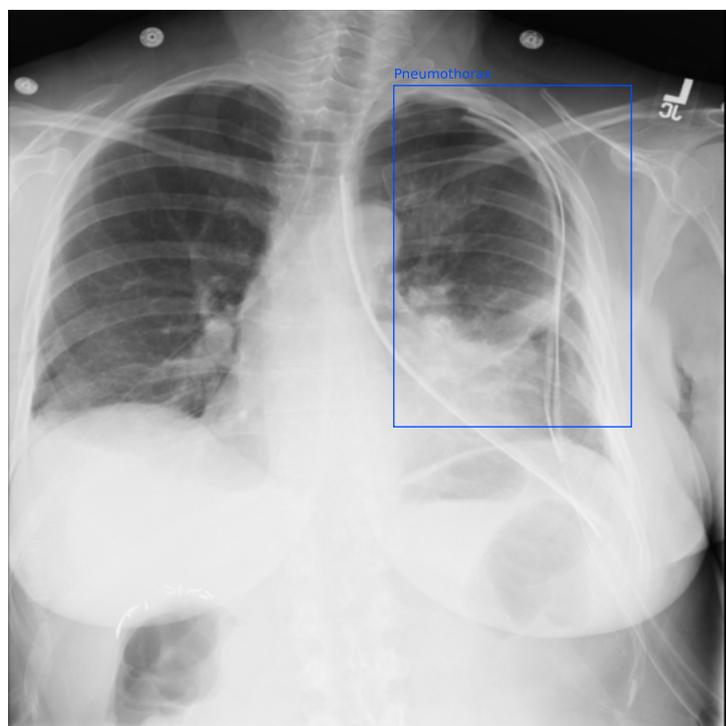


Figura 33: Neumotórax.

4 Metodología

En este capítulo se describe la metodología empleada para el desarrollo del proyecto. Primero, se describe lo relacionado con los datos como la partición de los subconjuntos, el preprocesamiento y las estrategias de aumento de datos empleadas. Después, se describe el esquema de experimentación y proceso de evaluación. Finalmente, se detallan los recursos de hardware y software utilizados.

4.1 Conjunto de datos

Para realizar la partición de los datos primero se llevó a cabo un análisis del conjunto. Se extrajeron estadísticas y gráficas para visualizar las características y distribuciones de las dos versiones del conjunto presentadas en el Capítulo 3.

4.1.1. Partición

Para el entrenamiento de los modelos en este proyecto se realizó una triple partición del conjunto de datos con los subconjuntos de entrenamiento, validación y prueba. La manera en que se generan los subconjuntos repercute directamente en el desempeño del modelo a entrenar. Por lo tanto, lo apropiado es que cada uno de los subconjuntos siga la distribución del conjunto global. Sin embargo, un error común es que al generar la división, los subconjuntos tengan una distribución diferente a la original en el conjunto.

Para asegurar la correcta generación de los subconjuntos se utilizó el algoritmo de *partición iterativa estratificada* [53]. Con este algoritmo se realizó la generación de los subconjuntos para las versiones reducida y completa descritas en el Capítulo 3. Los subconjuntos para ambas versiones quedaron establecidos como 70% entrenamiento, 15% validación y 15% prueba. El Cuadro 3 muestra la distribución del número de ejemplos en los subconjuntos para la versión reducida y completa.

	Entrenamiento	Validación	Prueba
Versión Completa	36,224 (70%)	7,709 (15%)	7,826 (15%)
Versión Reducida	611 (69%)	136 (16%)	133 (15%)

Cuadro 3: Número de ejemplos en los subconjuntos generados mediante estratificación iterativa para la versión completa y reducida.

El número de ejemplos por clase para los subconjuntos en las versiones completa y reducida se muestran en los Cuadros 4 y 5 respectivamente.

	atelectasia	cardiomegalia	consolidación	edema	derrame	enfisema	fibrosis	hernia	infiltración	masa	nódulo	enr. pleural	neumonía	neumotórax
Total	11559	2776	4667	2303	13317	2516	1686	227	19894	5782	6331	3385	1431	5302
Entrenamiento	8091	1943	3267	1612	9322	1761	1180	159	13926	4047	4432	2369	1002	3711
Validación	1734	417	700	345	1997	378	253	34	2984	868	950	508	215	795
Prueba	1734	416	700	346	1998	377	253	34	2984	867	949	508	214	796

Cuadro 4: Número de ejemplos por clase para los subconjuntos de la versión completa.

	atelectasia	cardiomegalia	derrame	infiltración	masa	nódulo	neumonía	neumotorax
Total	180	146	153	123	85	79	120	98
Entrenamiento	126	102	107	86	59	55	84	69
Validación	27	22	23	19	13	12	18	15
Prueba	27	22	23	18	13	12	18	14

Cuadro 5: Número de ejemplos por clase para los subconjuntos de la versión reducida.

4.1.2. Procesamiento

El preprocesamiento del conjunto de datos consistió en escalar las imágenes de 1024×1024 a dos resoluciones distintas, 512×512 y 256×256 para ambas versiones del conjunto de datos. Para realizar el redimensionado de las imágenes se empleó el método de interpolación bilineal, pues brinda un equilibrio entre la distorsión visual que produce y el tiempo de cómputo requerido [54].

4.1.3. Aumentado de datos

El aumentado de datos es una técnica que se emplea en el entrenamiento y consiste en aplicar transformaciones a los ejemplos de entrenamiento para generar más datos. Las transformaciones que se pueden aplicar a los datos dependen directamente de la modalidad de los datos (imágenes, audio, video o texto) que estemos analizando y además del dominio (imágenes médicas, audios de animales o videos de acciones humanas.). En el caso de las imágenes, como se muestra en [55], las transformaciones se pueden categorizar en:

- Transformaciones geométricas: giro, rotación, corte, translación, etc.
- Espacios de color: cambiar los canales de color, modificar contraste, intensificar color, etc.
- Filtros: desenfoque, adición de ruido gaussiano, etc.
- Borrado aleatorio: borrar una sección pequeña de la imagen.

El aumentado de datos es una técnica que no solo incrementa el número de ejemplos, también aumenta la diversidad del conjunto de datos. Esta técnica ayuda a reducir el sobreajuste [56].

Para aplicar aumentado de datos se pueden elegir entre dos estrategias. La primera consiste en aplicar las transformaciones antes del entrenamiento del modelo y almacenarlas como parte del conjunto de datos original. Esta estrategia se puede realizar cuando no se cuenta con muchos datos y no representa un problema el almacenarlos. En la segunda, las transformaciones a las imágenes se realizan durante el entrenamiento antes de introducirlas al modelo, en esta estrategia

no se requiere de espacio extra de almacenamiento. En este proyecto se realizó el proceso de aumentado de datos utilizando la segunda estrategia descrita.

Se ha estudiado sobre las transformaciones que favorecen el entrenamiento de modelos basados en redes neuronales para resolver problemas de clasificación usando radiografías de tórax sin distorsionar los patrones anormales visuales [57]. Con base en esa evidencia, se aplicaron a las imágenes las siguientes transformaciones de manera aleatoria:

- Recorte de bordes aleatorio entre los valores de 25,37 y 50 píxeles. La imagen recortada se redimensiona al tamaño original.
- Rotación respecto del centro de la imagen con valor aleatorio entre $(-n, n)$ donde $n \in \{0^\circ, 15^\circ, 30^\circ, 45^\circ, 60^\circ\}$.
- Escalado de la imagen con valor aleatorio dentro del rango $(0.85, 1.15)$. Tomando 1 como el tamaño original de la imagen.
- Translación de la imagen en horizontal y vertical con un valor de movimiento aleatorio dentro de un rango dado en relación con las dimensiones de la imagen.

4.2 Esquema de experimentación

La experimentación en este proyecto contempló el desarrollo de modelos de clasificación multi-etiqueta de patologías asociadas a radiografías de tórax con base en evidencia visual de patrones anormales. Para el desarrollo de los modelos se contemplaron diversos aspectos con el propósito de analizar el impacto en el desempeño de los modelos de clasificación. Uno de los aspectos fue la arquitectura de la red convolucional. Se usaron cuatro arquitecturas del estado del arte, ShuffleNet [58], DenseNet [43], MobileNet [59] y ResNet [19].

Para la versión reducida del conjunto se realizó experimentación usando aumentado de datos. Además, se probó el impacto de la resolución de las imágenes en el desempeño, con resoluciones de 224, 256, 512 y 1024.

También se usó transferencia de conocimiento empleando modelos preentrenados en los conjuntos ImageNet [60], MIMIC-CXR [61], CheXpert [62], PadChest [63]. Para la adaptación de la arquitectura, se reusaron las capas preentrenadas de la etapa de extracción de características y se reemplazó la etapa de clasificación por una capa completamente conectada con el número de salidas requerido. Al ser un problema de clasificación multi-etiqueta, cada salida se pasa por una función *sigmoide*. Se experimentó utilizando el bloque convolucional como extractor de características y también se experimentó reentrenándolo. Primero, la transferencia de conocimiento solo se aplicó utilizando la versión reducida del conjunto. Después, con la configuración del modelo con mejor desempeño, se realizó experimentación con la versión completa. El optimizador utilizado en el entrenamiento fue *Adam* [35]. La Figura 34 muestra el flujo de predicción para clasificación.

como se observa en la curva morada de la Figura 35. La línea punteada de comportamiento lineal en dicha figura representa un clasificador aleatorio, mientras que la curva del peor clasificador se muestra justo por encima de la línea punteada.

El área bajo la curva (AUC, por sus siglas en inglés) mide el área bidimensional completa debajo de la curva ROC. El AUC proporciona una medida agregada del rendimiento en todos los umbrales de clasificación posibles.

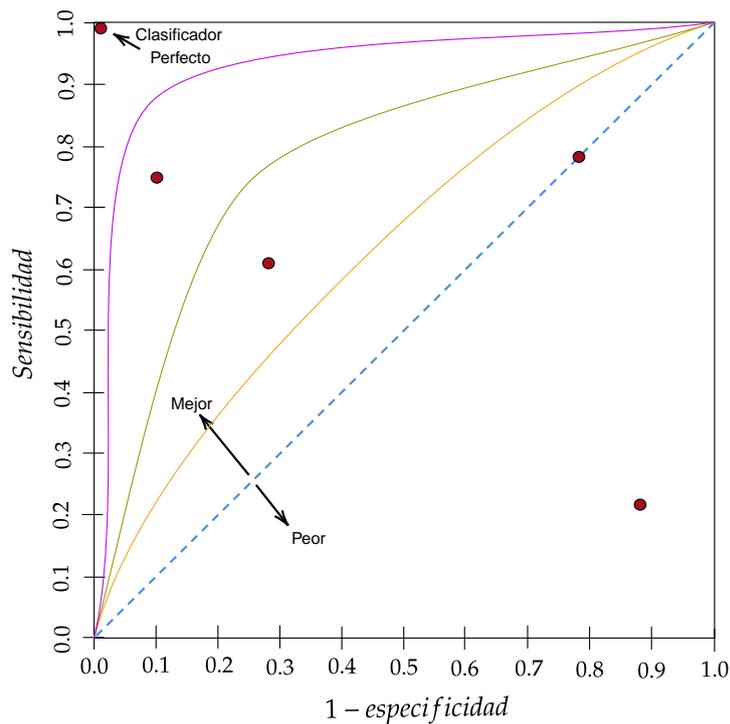


Figura 35: Curvas ROC para distintos clasificadores.

4.3.2. Métricas multietiqueta

Las métricas de clasificación multietiqueta son un indicador de desempeño a través de todas las categorías sobre todos los ejemplos y sus predicciones. Para este proyecto se eligieron cuatro métricas que agrupan los desempeños de AUC en las clases. Las métricas ocupadas fueron los promedios micro μAP , macro mAP , ponderado wAP y por instancia iAP .

El promedio micro μAP se calcula usando todas las etiquetas como si se tratara de una única tarea de clasificación binaria. En esta métrica, las clases que tienen un mayor número de ejemplos asociados tiene un mayor impacto. Por lo tanto, si los ejemplos asociados a las clases con mayor presencia son bien clasificadas, el resultado del promedio micro será alto. El promedio macro mAP se obtiene de calcular el AUC para cada clase y después se realiza un promedio. A diferencia del promedio micro, esta métrica no depende del número de ejemplos asociados a las clases.

Para obtener el promedio ponderado wAP se calcula de la misma manera que el promedio macro y además se multiplica por la frecuencia de la clase. El promedio por instancia iAP se obtiene calculando el AUC por cada instancia y después se realiza un promedio.

4.4 Recursos de hardware y software

La implementación se realizó en el lenguaje de programación Python. Los editores empleados fueron *Atom* y *GNU Nano*. Para la organización de recursos utilizados se trabajó con el manejador de paquetes y ambientes *Anaconda*. Las bibliotecas de cómputo científico y aprendizaje de máquinas fueron NumPy, scikit-learn y PyTorch. La ejecución de los entrenamientos y el registro de las métricas de evaluación se realizaron con Neptune.ai, que cuenta con una plataforma para monitoreo en tiempo real.

El hardware utilizado en este proyecto involucra tres sistemas en específico: los servicios de cómputo en la nube de Google, un servidor del Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas (IIMAS) de la Universidad Nacional Autónoma de México (UNAM), y una computadora personal. La computadora personal fue la principal herramienta de desarrollo de código. Para la ejecución de las pruebas desarrolladas se emplearon los servicios de cómputo en la nube (*Colaboratory*). La ejecución de los experimentos se realizó en el servidor del IIMAS debido a su capacidad de procesamiento. El Cuadro 6 describe los recursos de hardware utilizados en el desarrollo del proyecto.

Sistema	Sist. Operativo	CPU	Mem. RAM	GPU	GPU RAM	Almac. Ocupado
PC	Windows 11	Intel Core i7 7Gen	16 GB	GTX 1060	6 GB	50 GB
Nube Google	Ubuntu 18.04 LTS	Intel Xeon	12 GB	Tesla P100-PCIE	16 GB	50 GB
Servidor IIMAS	Ubuntu 20.04 LTS	Intel Core i9 10Gen	128 GB	RTX 3090	24 GB	95 GB

Cuadro 6: Recursos de hardware utilizados en el proyecto.

5 Análisis de resultados

La experimentación que se llevó a cabo se describe en el Capítulo 4. Esta experimentación busca estudiar el impacto de varios aspectos en la configuración del modelo para la clasificación multietiqueta de patologías. La primera línea de experimentación consistió en estudiar la influencia del aumento de datos. En la segunda línea se estudió el impacto de la arquitectura de la red neuronal convolucional. Otro aspecto que se estudió en la tercera línea de experimentación fue el impacto de la transferencia de conocimiento. Finalmente, la última línea consistió en una comparación de la mejor configuración entre la versión reducida y completa de los datos. Los resultados de cada una de las líneas de experimentación se muestran en las siguientes secciones.

Las tres primeras líneas se realizaron con la versión reducida del conjunto de datos. Posteriormente, con la configuración del modelo con que se obtuvieron los mejores resultados en la versión reducida, se experimentó en la versión completa.

5.1 Aumentado de datos

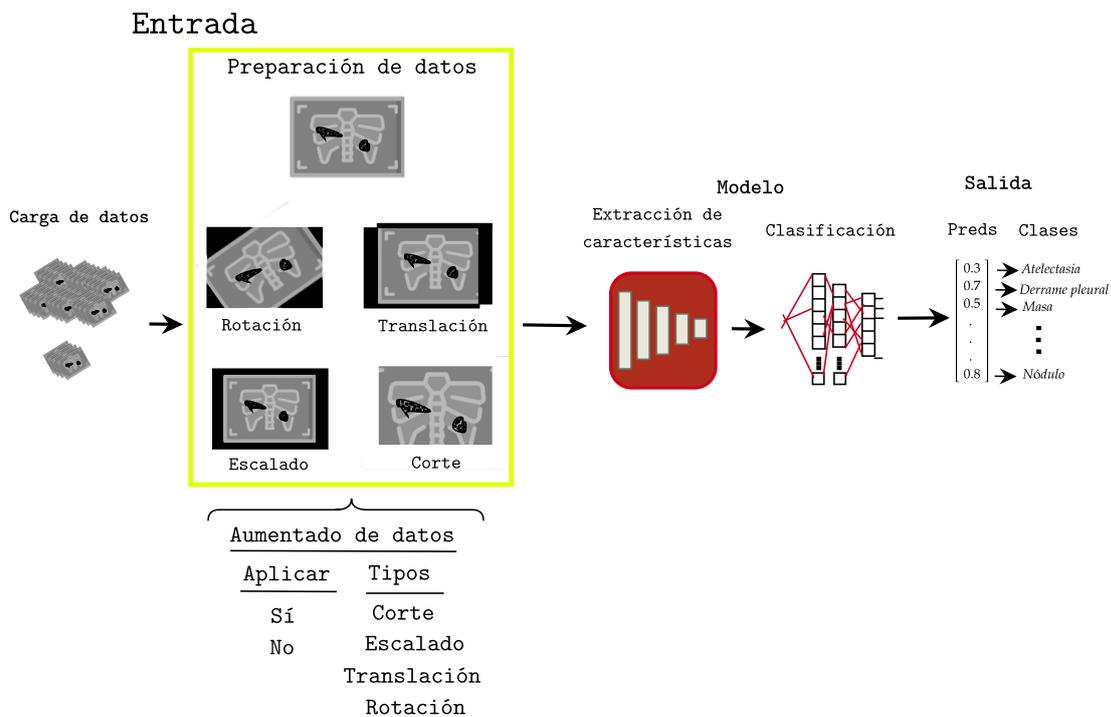


Figura 36: Diagrama de experimentación para estudiar el impacto del aumento de datos sobre la versión reducida de los datos.

Para la experimentación se utilizó la arquitectura *DenseNet121* [43] sin preentrenamiento. Primero, como modelo de referencia se entrenó la arquitectura sin ningún tipo de aumentado de datos. Después, se entrenaron una serie de modelos, cada uno con una técnica de aumentado de datos por separado. Finalmente, se entrenó un modelo aplicando todas las técnicas de aumentado anteriores. La Figura 36 ilustra la experimentación que realizada para ver el impacto del aumentado.

Las transformaciones que se aplicaron fueron descritas a detalle en el Capítulo 3. El corte aleatorio se efectúa en el alto y ancho de la imagen. El escalado se encarga de modificar la escala visual de la radiografía, cortándola o agrandándola en porcentajes aleatorios respecto de las dimensiones originales. La translación mueve el centro de la imagen cierto número de píxeles horizontal y verticalmente dentro de un rango. La rotación gira la imagen de manera aleatoria dentro de un rango definido.

Los resultados obtenidos acorde a las métricas definidas en la Sección 4.3.2, se encuentran en el Cuadro 7. Las métricas muestran que el modelo entrenado sin aumentado de datos y aplicando todas las técnicas tiene desempeños muy similares.

Aumentado	<i>uAP</i> (%)	<i>mAP</i> (%)	<i>wAP</i> (%)	<i>iAP</i> (%)
Sin Aumentado	65.40	63.04	63.46	64.09
Corte Al.	62.48	60.77	61.31	63.10
Rotación	59.54	58.42	59.22	58.74
Translación	63.50	61.69	62.50	62.23
Escalado	63.31	61.71	61.83	66.07
Todas	65.33	63.11	63.91	65.98

Cuadro 7: Resultados para el modelo usando aumentado de datos.

El Cuadro 8 muestra los resultados de AUC-ROC para cada patología. El modelo sin aumentado de datos obtuvo mejor desempeño para las patologías de *atelectasia*, *nódulo* y *neumotórax*. Sin embargo, se muestra una mejora en desempeño combinando todas las técnicas de aumentado de datos para las patologías de *derrame pleural*, *masa* y *neumonía*. El modelo que incrementa los datos realizando cortes aleatorios mostró el mejor desempeño para *infiltración*. Mientras que el modelo que usa el escalado se desempeña mejor en la patología de *cardiomegalia*.

Los resultados a nivel patología sugieren que algunas patologías son muy sensibles a las técnicas de aumentado de datos. En algunos casos, aplicar una transformación a una patología puede dificultar su reconocimiento. Un ejemplo de esto es la categoría de *neumotórax*, donde el corte aleatorio afecta drásticamente el desempeño en comparación a no aplicar aumentado de datos o al aplicar el escalado. Ninguna estrategia de aumentado por sí sola mejora el desempeño de todas las patologías. Sin embargo, todas en conjunto tienen desempeños similares a entrenar con el conjunto original.

Aumentado	atelectasia	cardiomegalia	derrame	infiltración	masa	nódulo	neumonía	neumotórax
Sin Aumentado	66.15	78.77	58.06	65.17	52.76	67.93	44.11	71.38
Corte Aleatorio	65.24	77.67	52.70	66.78	54.90	64.06	45.12	59.68
Rotación	62.76	67.60	60.85	59.23	40.58	66.52	46.98	62.82
Translación	65.44	75.32	60.40	62.73	47.61	65.47	51.43	65.13
Escalado	61.71	83.95	50.79	60.95	58.12	62.23	46.24	69.71
Todas	62.76	78.02	66.99	61.17	59.12	58.20	54.69	63.91

Cuadro 8: Resultados en porcentaje (%) del AUC-ROC por patología usando aumentado de datos.

5.2 Arquitectura

La Figura 37 ilustra la experimentación realizada en el análisis de las arquitecturas. Se entrenaron diferentes arquitecturas convolucionales notables. Para todos los modelos se eligieron los mismos hiperparámetros y además se aplicó aumentado de datos. Las transformaciones usadas para esta ronda de experimentos fueron rotaciones aleatorias de máximo 15 grados y cortes aleatorios, como una estrategia para contrarrestar el sobreajuste.

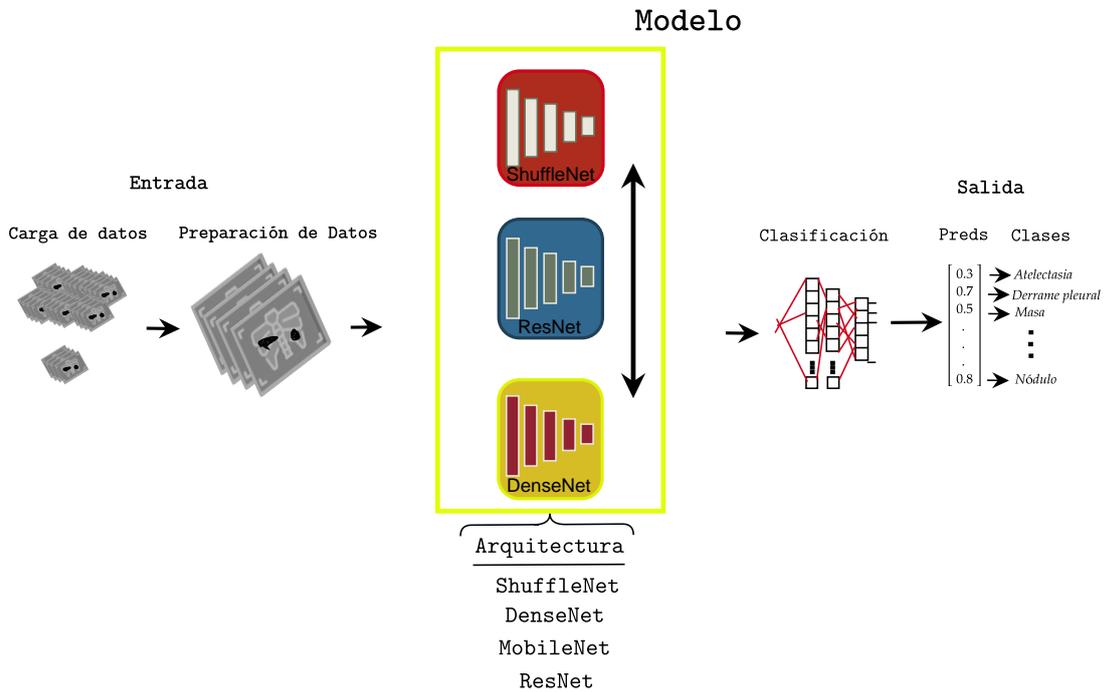


Figura 37: Diagrama de experimentación para estudiar el impacto de la arquitectura sobre la versión reducida de los datos.

Las arquitecturas convolucionales pueden dividirse en dos grandes grupos, enfocadas en desempeño y enfocadas en eficiencia desde el punto de vista del número de parámetros que emplean.

Para el desarrollo de la experimentación se usaron ambos tipos de arquitecturas. Por el lado eficiente se usó *ShuffleNet* [58] y *MobileNetv2* [59]. *ShuffleNet* en sus dos versiones *ShuffleNetv2x05* y *ShuffleNetv2x10*. Por el lado enfocado en desempeño se usó *ResNet* [19] y *DenseNet* [43]. *ResNet* en sus versiones *ResNet34*, *Resnet50*, *ResNet101* y *ResNet152*. Mientras que *DenseNet* en sus versiones *DenseNet121*, *DenseNet161*, *DenseNet169* y *DenseNet201*.

Arquitectura	<i>uAP</i> (%)	<i>mAP</i> (%)	<i>wAP</i> (%)	<i>iAP</i> (%)
<i>ShuffleNetv2x05</i>	66.23	65.37	65.92	66.63
<i>ShuffleNetv2x10</i>	64.70	64.06	64.33	66.56
<i>MobileNetv2</i>	68.19	65.86	67.08	68.12
<i>ResNet18</i>	62.21	64.05	64.08	62.35
<i>ResNet34</i>	64.81	63.91	64.30	65.12
<i>ResNet50</i>	63.21	62.38	63.31	62.61
<i>ResNet101</i>	64.11	63.72	63.84	63.30
<i>ResNet152</i>	60.86	60.19	60.47	59.62
<i>DenseNet121</i>	68.05	66.88	67.46	67.78
<i>DenseNet161</i>	62.77	62.90	63.21	61.21
<i>DenseNet169</i>	67.69	65.74	65.94	67.22
<i>DenseNet201</i>	60.37	60.35	61.21	60.20

Cuadro 9: Resultados para el modelo usando diferentes arquitecturas convolucionales.

Arquitectura	atelectasia	cardiomegalia	derrame	infiltración	masa	nódulo	neumonía	neumotórax
<i>ShuffleNetv2x05</i>	64.29	85.08	62.65	60.34	50.80	66.88	56.80	76.15
<i>ShuffleNetv2x10</i>	67.26	79.01	58.22	58.62	48.09	72.81	50.60	77.85
<i>MobileNetv2</i>	66.65	88.48	64.85	69.46	54.64	57.97	53.84	71.03
<i>ResNet18</i>	57.88	80.01	62.60	65.35	57.05	68.91	56.50	64.10
<i>ResNet34</i>	62.26	83.63	60.98	54.25	58.56	64.14	61.73	65.71
<i>ResNet50</i>	61.11	88.22	61.47	65.48	50.83	66.45	49.42	56.03
<i>ResNet101</i>	54.80	79.62	66.70	66.81	53.39	69.34	55.40	63.69
<i>ResNet152</i>	57.62	75.79	62.38	55.60	45.06	68.44	45.81	70.80
<i>DenseNet121</i>	65.18	87.61	63.31	62.81	50.09	69.73	60.69	75.64
<i>DenseNet161</i>	59.09	79.15	66.81	55.26	57.45	64.65	53.65	67.15
<i>DenseNet169</i>	61.69	89.00	52.53	63.07	45.28	71.91	69.85	72.60
<i>DenseNet201</i>	60.77	81.00	60.71	61.43	54.53	57.54	45.12	61.73

Cuadro 10: Resultados en porcentaje (%) del AUC-ROC por patología usando diferentes arquitecturas convolucionales.

En el Cuadro 9 se muestran los resultados obtenidos con las diferentes arquitecturas. La arquitectura eficiente *MobileNetv2* obtiene los mejores resultados para μap y iAP . El resultado de *DenseNet121* en las métricas wAP y mAP es el más alto.

En el Cuadro 10 se muestran los resultados de AUC para las patologías. *ShuffleNetv2x10* tiene

un desempeño para las patologías de *atelectasia*, *nódulo* y *neumotórax*. Sin embargo, patologías como *cardiomegalia*, *derrame pleural*, *masa* y *neumonía* se ven beneficiadas de arquitecturas enfocadas en desempeño.

5.3 Preentrenamiento

Para estudiar como influye el preentrenamiento se utilizó la arquitectura *DenseNet121* preentrenada de manera aislada en los conjuntos de datos *ImageNet* [60], *MIMIC-CXR* [61], *CheXpert* [62] y *PadChest* [63]. Esto se ilustra en la Figura 38.

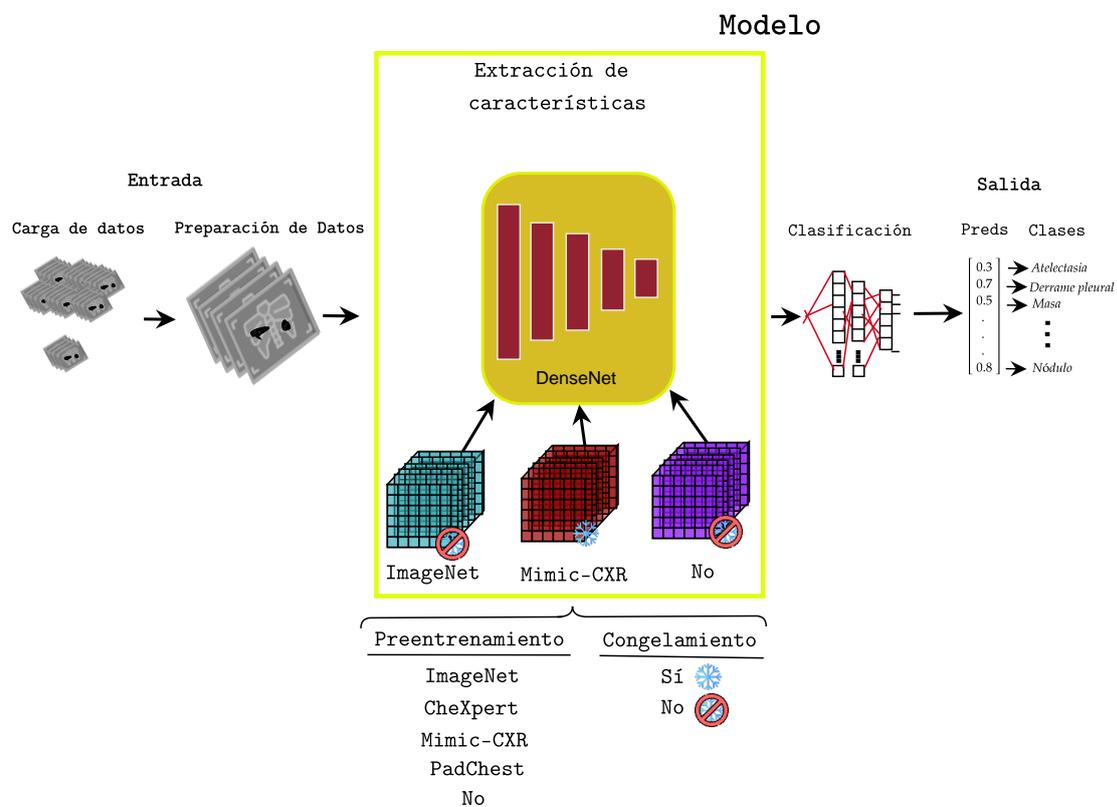


Figura 38: Diagrama de experimentación para estudiar el impacto de la transferencia de conocimiento sobre la versión reducida de los datos.

Primero se realizó la transferencia de conocimiento utilizando la red *DenseNet121* como un extractor de características. Los parámetros de las capas convolucionales fueron congelados y solo se reentrenó la red completamente conectada. En la segunda etapa de experimentación se tomó el modelo que presentó mejores resultados y se afinaron los parámetros de las capas convolucionales.

En el Cuadro 11 se muestran los resultados obtenidos con los distintos modelos preentrenados. El modelo que no tiene preentrenamiento obtuvo los resultados más bajos. Esto se debe a que el

conjunto de datos cuenta con muy pocos ejemplos y las radiografías son imágenes complejas, lo cual implica que se requiera mayor cantidad de datos para mejorar el ajuste de los modelos. Dentro de los modelos preentrenados, el que tuvo peor desempeño fue el preentrenado en *ImageNet*. Esto sugiere que el modelo se beneficia de tener un preentrenamiento en un conjunto de radiografías, pues los patrones de las radiografías son muy específicos.

El modelo preentrenado en *PadChest* obtuvo los mejores resultados. Esto se debe a que *PadChest* es uno de los conjuntos más grandes de radiografías dentro de los que se preentrenó. Además, posee imágenes y etiquetado de calidad, lo que permite aprender mejores representaciones. Un modelo con desempeño similar al preentrenado en *PadChest* es el preentrenado en *CheXpert*.

Los resultados obtenidos afinando los parámetros de la red convolucional fueron inferiores en comparación a usarla como extractor de características.

Preentrenamiento	<i>uAP</i> (%)	<i>mAP</i> (%)	<i>wAP</i> (%)	<i>iAP</i> (%)
Extractor				
Ninguno	62.47	60.28	60.77	61.74
<i>ImageNet</i>	63.51	66.01	65.90	61.81
<i>CheXpert</i>	70.75	67.85	68.05	71.23
<i>MIMIC-CXR</i>	66.82	64.40	65.43	65.25
<i>PadChest</i>	71.14	68.73	68.75	71.04
Afinado				
<i>PadChest</i>	68.96	66.06	66.93	68.54

Cuadro 11: Resultados para el modelo usando transferencia de conocimiento.

Los resultados de AUC a nivel patología se muestran en el Cuadro 12. Los resultados para las patologías mayoritariamente incrementan en comparación con los resultados obtenidos en las secciones anteriores. En su mayoría, los mejores resultados a nivel patología se obtienen con el modelo preentrenado de *PadChest* y *CheXpert*.

Preentrenamiento	atelectasia	cardiomegalia	derrame	infiltración	masa	nódulo	neumonía	neumotórax
Extractor								
Ninguno	60.00	69.13	65.58	61.11	53.24	66.76	51.37	55.03
<i>ImageNet</i>	58.53	82.36	68.54	64.40	58.31	74.14	51.12	70.67
<i>CheXpert</i>	60.22	86.98	69.81	69.14	62.60	70.90	57.51	65.61
<i>MIMIC-CXR</i>	67.16	81.42	68.50	65.75	49.09	74.61	51.70	56.99
<i>PadChest</i>	64.68	92.17	66.16	61.59	64.26	78.95	55.92	66.12
Afinado								
<i>PadChest</i>	64.13	86.09	68.85	69.06	57.64	65.43	53.23	64.07

Cuadro 12: Resultados en porcentaje (%) del AUC-ROC por patología con transferencia de conocimiento.

De manera general, se puede concluir que para la clasificación de patologías se obtienen mejores resultados utilizando la técnica de transferencia de conocimiento.

5.4 Versiones reducida y completa

Como se mencionó en el Capítulo 3 la diferencia entre la versión reducida y completa no sólo es el número de ejemplos, sino también el número de clases. La Figura 39 ilustra la experimentación realizada para analizar el impacto del conjunto de datos en el desempeño del modelo.

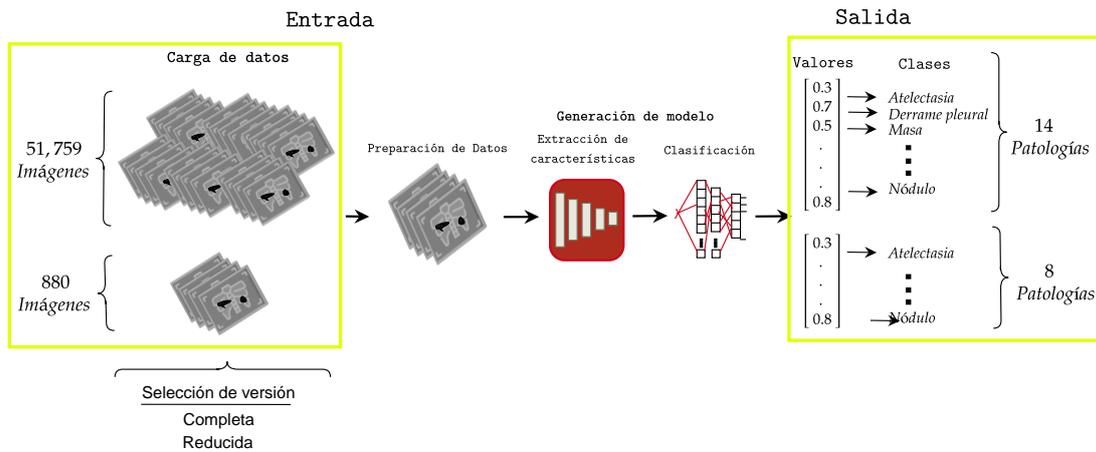


Figura 39: Diagrama de experimentación para estudiar el impacto del tamaño del conjunto de datos en el modelo de clasificación .

Para realizar esta experimentación se empleó la mejor configuración del modelo con base en los resultados obtenidos de las secciones anteriores. Se hizo uso de la técnica de transferencia de conocimiento con la arquitectura *DenseNet121* preentrenada en *PadChest* usando la combinación de las cuatro técnicas de aumentado de datos presentadas en la Sección 5.1.

Los resultados obtenidos se muestran en el Cuadro 13. El mejor desempeño del modelo se obtiene utilizando la versión completa. Esto se puede explicar debido a que esta versión posee, por mucho, un mayor número de ejemplos. Este mismo efecto se ve en los resultados de AUC para las patologías que se muestran en el Cuadro 14. Las dos únicas patologías para las cuales el desempeño fue inferior usando la versión completa fueron *nódulo* y *cardiomegalia*. Esto se debe a que para ambas categorías el porcentaje de imágenes asociadas respecto al total es superior en la versión reducida.

5.5 Configuración final

La Figura 40 muestra la mejor configuración con base en las líneas de experimentación antes presentadas.

Versión	<i>uAP</i> (%)	<i>mAP</i> (%)	<i>wAP</i> (%)	<i>iAP</i> (%)
Reducida	71.24	68.61	68.63	70.26
Completa	82.66	73.58	71.35	82.86

Cuadro 13: Resultados del modelo con la mejor configuración para la versión completa y reducida.

Versión	atelectasia	cardiomegalia	derrame	infiltración	masa	nódulo	neumonía	neumotórax	consolidación	edema	efisema	fibrosis	hernia	Engr. Pleural
Reducida	64.48	92.12	65.76	61.91	63.56	78.98	56.20	65.87	–	–	–	–	–	–
Completa	70.99	89.26	78.03	65.84	71.45	67.79	60.06	72.66	68.03	80.34	69.72	74.76	91.37	69.79

Cuadro 14: Resultados en porcentaje (%) del conjunto de prueba en ambas versiones del conjunto de datos.

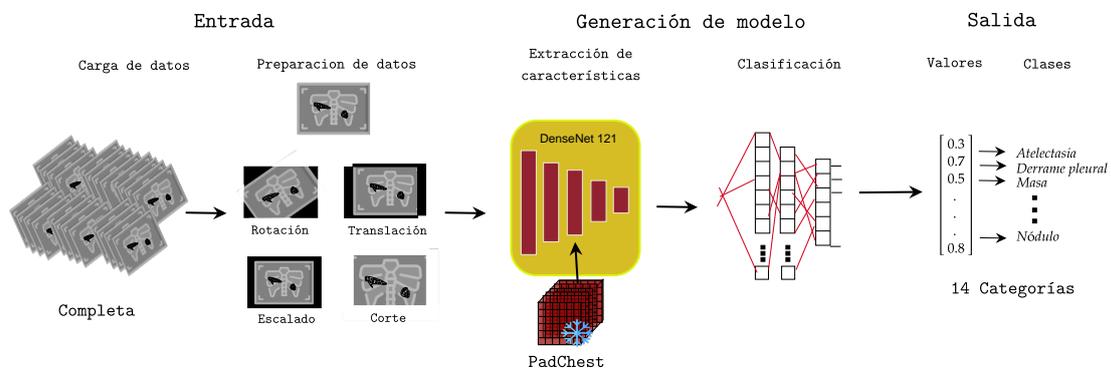
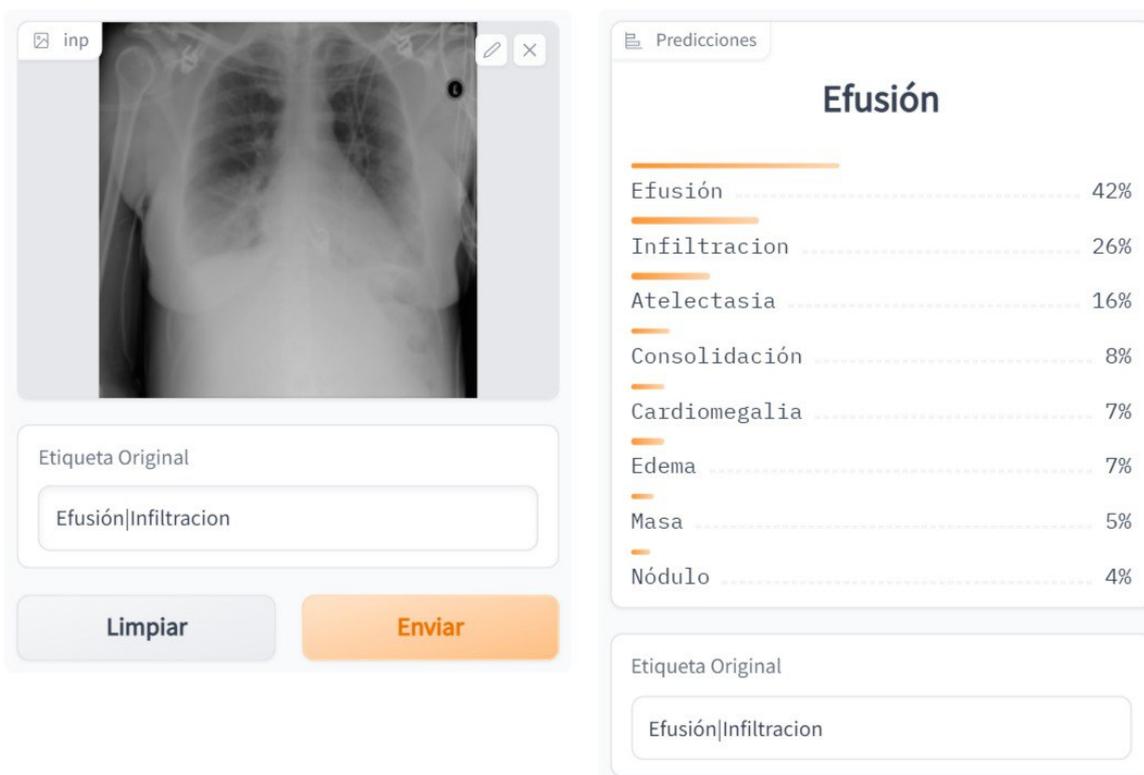


Figura 40: Mejor configuración del modelo de clasificación de patologías.

En esta Sección se presenta la mejor configuración encontrada para modelo de clasificación de patologías y se discute cada uno de sus elementos. El mejor modelo con base en la experimentación se compone de la arquitectura *DenseNet121* preentrenada en *PadChest* usando aumentado de datos y siendo entrenada con la versión completa de los datos. Las Figuras 41 y 42 ilustran algunas de las predicciones realizadas por este modelo.

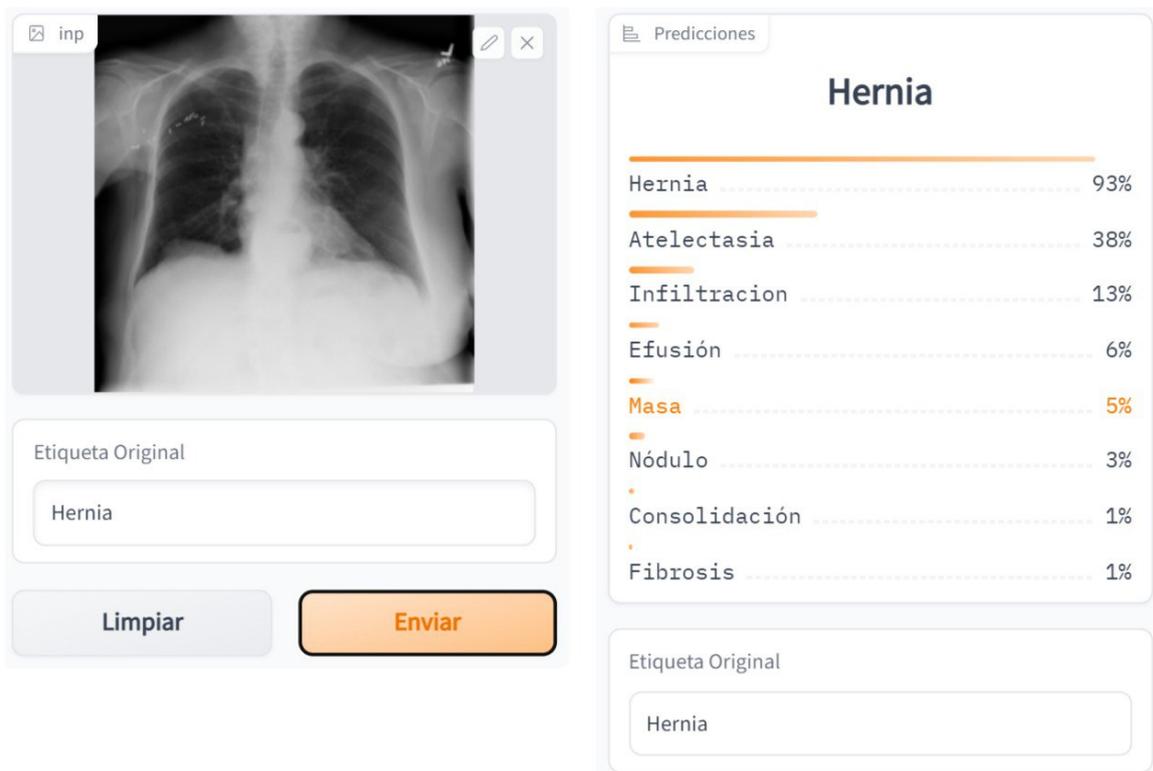


(a) Predicción multietiqueta: 2 patologías



(b) Predicción multietiqueta: 4 patologías

Figura 41: Predicciones usando la configuración del modelo con mejor desempeño en la métrica uAP.



(a) Predicción de patología: Hernia

Figura 42: Predicciones usando la configuración del modelo con mejor desempeño en la metrica uAP.

6 Conclusiones

6.1 Conclusiones

En este proyecto se abordó la tarea de clasificación multietiqueta de patologías a partir de radiografías de tórax usando redes neuronales convolucionales por su capacidad de aprender patrones complejos y discriminatorios de las imágenes. Para derivar el modelo final se estudió el impacto en el desempeño de aspectos como aumento de datos, arquitectura, preentrenamiento y tamaño del conjunto de datos. Para dar solidez a la experimentación, se realizó una partición estratificada de los datos en subconjuntos de entrenamiento, validación y prueba manteniendo la distribución original de las clases. Los modelos fueron evaluados en la partición de prueba con el objetivo de obtener resultados más generalizables.

De los resultados obtenidos es posible concluir que los factores que más influyen en el desempeño del modelo son el conjunto usado para el preentrenamiento y el tamaño del conjunto de datos. En el caso del preentrenamiento, los resultados muestran que una arquitectura preentrenada con ImageNet supera el desempeño de una inicialización aleatoria. A su vez, el preentrenamiento con los conjuntos de imágenes de rayos X de tórax, supera las dos primeras estrategias de inicialización. El modelo con preentrenamiento en PadChest supera 8.67 y 7.63 puntos de μAP la inicialización aleatoria y preentrenamiento en ImageNet respectivamente. Estos resultados son acorde a la literatura de transferencia de conocimiento [65] en los que se ha encontrado que el preentrenamiento en conjuntos de datos similares al conjunto objetivo produce modelos con mayor desempeño a conjuntos de preentrenamientos de naturaleza distinta o inicialización aleatoria. Por otra parte, los resultados obtenidos señalan que el entrenamiento en la versión completa del conjunto (≈ 59 veces más grande que la versión reducida), produce el modelo con mejor desempeño (11.42 % puntos de incremento en μAP). El incremento de desempeño de las redes neuronales con respecto al incremento del tamaño del conjunto de datos de entrenamiento es un fenómeno que se ha comprobado en trabajos en clasificación de imágenes naturales [66], así como también en el procesamiento de lenguaje natural [67].

6.2 Trabajo a futuro

Con base en la experiencia y los resultados obtenidos durante el desarrollo de este proyecto, las siguientes líneas de trabajo a futuro se consideran interesantes.

- Desarrollar modelos considerando información clínica del paciente. Para realizar un diagnóstico, los radiólogos toman en cuenta información clínica del paciente adicionalmente a la radiografía. Desarrollar modelos que integren esta información puede ayudar a mejorar su desempeño y confiabilidad.

- **Validación médica.** Un paso importante en la evaluación final de este tipo de herramientas es la validación por especialistas en el área. Los profesionales pueden orientar a corregir los posibles sesgos o errores que estos modelos llegan a presentar. Aunque esto requiere de una logística y coste de desarrollo mucho mayor. Sin embargo, el beneficio en la confiabilidad es invaluable.
- **Interfaz de usuario.** El acercamiento de este tipo de modelos a los usuarios finales es un paso muy importante en su utilidad. Por lo cual, el desarrollo de interfaces de usuario que permitan un uso fácil y práctico es fundamental para el avance de este tipo de sistemas en el uso cotidiano.

7 Referencias

- [1] P. Suetens, *Fundamentals of medical imaging*. Cambridge university press, 2017.
- [2] A. Krizhevsky, I. Sutskever y G. E. Hinton, «ImageNet Classification with Deep Convolutional Neural Networks,» *Commun. ACM*, vol. 60, n.º 6, págs. 84-90, mayo de 2017, ISSN: 0001-0782.
- [3] W. Zhiqiang y L. Jun, «A review of object detection based on convolutional neural network,» en *2017 36th Chinese Control Conference (CCC)*, 2017, págs. 11 104-11 109.
- [4] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou y G. Cottrell, «Understanding Convolution for Semantic Segmentation,» en *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018, págs. 1451-1460.
- [5] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta y A. A. Bharath, «Generative Adversarial Networks: An Overview,» *IEEE Signal Processing Magazine*, vol. 35, n.º 1, págs. 53-65, 2018.
- [6] J. Yang, R. Shi y B. Ni, «MedMNIST Classification Decathlon: A Lightweight AutoML Benchmark for Medical Image Analysis,» en *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, 2021, págs. 191-195.
- [7] G. Maguolo y L. Nanni, «A critic evaluation of methods for COVID-19 automatic detection from X-ray images,» *Information Fusion*, vol. 76, págs. 1-7, 2021, ISSN: 1566-2535.
- [8] A. T. Sahlol, A. M. Abdeldaim y A. E. Hassanien, «Automatic acute lymphoblastic leukemia classification model using social spider optimization algorithm,» *Soft Computing*, vol. 23, págs. 6345-6360, 2019.
- [9] «Machine-learning-based multiple abnormality prediction with large-scale chest computed tomography volumes,» *Medical Image Analysis*, vol. 67, pág. 101 857, 2021, ISSN: 1361-8415.
- [10] S. R. van der Voort, M. Smits, S. Klein y A. D. N. Initiative, «DeepDicomSort: an automatic sorting algorithm for brain magnetic resonance imaging data,» *Neuroinformatics*, vol. 19, n.º 1, págs. 159-184, 2021.
- [11] The DeepRadiology Team, *Pneumonia Detection in Chest Radiographs*, 2018.
- [12] P. Neri, M. Fiaschi y G. Menchini, «Semi-Automatic Tool for Vitiligo Detection and Analysis,» *Journal of Imaging*, vol. 6, n.º 3, 2020, ISSN: 2313-433X.
- [13] S. J. Russell, *Artificial intelligence a modern approach*. Pearson Education, Inc., 2010.

- [14] J. C. Flowers, «Strong and Weak AI: Deweyan Considerations.,» en *AAAI spring symposium: Towards conscious AI systems*, vol. 2287, 2019.
- [15] A. Brahme, *Comprehensive biomedical physics*. Newnes, 2014, ISBN: 9780444536327.
- [16] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever y D. Amodei, «Language Models are Few-Shot Learners,» en *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan y H. Lin, eds., vol. 33, Curran Associates, Inc., 2020, págs. 1877-1901.
- [17] J. Devlin, M.-W. Chang, K. Lee y K. Toutanova, *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, 2018.
- [18] U. Shaham, E. Segal, M. Ivgi, A. Efrat, O. Yoran, A. Haviv, A. Gupta, W. Xiong, M. Geva, J. Berant y O. Levy, *SCROLLS: Standardized CompaRison Over Long Language Sequences*, 2022.
- [19] K. He, X. Zhang, S. Ren y J. Sun, «Deep Residual Learning for Image Recognition,» en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jun. de 2016.
- [20] A. Gordo, J. Almazán, J. Revaud y D. Larlus, «Deep Image Retrieval: Learning Global Representations for Image Search,» en *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe y M. Welling, eds., Cham: Springer International Publishing, 2016, págs. 241-257.
- [21] B. Zoph, V. Vasudevan, J. Shlens y Q. V. Le, «Learning Transferable Architectures for Scalable Image Recognition,» en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jun. de 2018.
- [22] H. Rebecq, R. Ranftl, V. Koltun y D. Scaramuzza, «Events-To-Video: Bringing Modern Computer Vision to Event Cameras,» en *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, jun. de 2019.
- [23] V. Fontana, G. Singh, S. Akrigg, M. Di Maio, S. Saha y F. Cuzzolin, *Action Detection from a Robot-Car Perspective*, 2018.
- [24] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.
- [25] M. J. Er, R. Venkatesan y N. Wang, «An online universal classifier for binary, multi-class and multi-label classification,» en *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2016, págs. 003 701-003 706.
- [26] A. Zhang, Z. C. Lipton, M. Li y A. J. Smola, «Dive into Deep Learning,» *arXiv preprint arXiv:2106.11342*, 2021.

- [27] D. E. Rumelhart, G. E. Hinton y R. J. Williams, «Learning internal representations by error propagation,» California Univ San Diego La Jolla Inst for Cognitive Science, inf. téc., 1985.
- [28] G. Wang y K. C. Sim, «Regression-Based Context-Dependent Modeling of Deep Neural Networks for Speech Recognition,» *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, n.º 11, págs. 1660-1669, 2014.
- [29] G. Chollet, M.-G. Di Benedetto, A. Esposito y M. Marinaro, *Speech Processing, Recognition and Artificial Neural Networks: Proceedings of the 3rd International School on Neural Nets “Eduardo R. Caianiello”*. Springer Science & Business Media, 2012.
- [30] M. Dib, *Automatic Speech Recognition of Arabic Phonemes with Neural Networks: A Contrastive Study of Arabic and English*. Springer, 2018.
- [31] G. Pineda, R. Montalvo y B. Montalvo, «Introducción al Aprendizaje Autosupervisado,» *Aprendizaje Profundo 2021-1, PCIC*, 2020.
- [32] C. M. Bishop, «Neural networks and their applications,» *Review of scientific instruments*, vol. 65, n.º 6, págs. 1803-1832, 1994.
- [33] F. Murtagh, «Multilayer perceptrons for classification and regression,» *Neurocomputing*, vol. 2, n.º 5, págs. 183-197, 1991, ISSN: 0925-2312.
- [34] I. Goodfellow, Y. Bengio y A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.
- [35] D. P. Kingma y J. Ba, «Adam: A method for stochastic optimization,» *arXiv preprint arXiv:1412.6980*, 2014.
- [36] G. Hinton, N. Srivastava y K. Swersky, «Lecture 6a overview of mini-batch gradient descent,» *Coursera Lecture slides <https://class.coursera.org/neuralnets-2012-001/lecture>*, 2012.
- [37] M. D. Zeiler, «ADADELTA: An Adaptive Learning Rate Method,» *CoRR*, vol. abs/1212.5701, 2012. arXiv: 1212.5701.
- [38] Z. Li, F. Liu, W. Yang, S. Peng y J. Zhou, «A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects,» *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, n.º 12, págs. 6999-7019, 2022.
- [39] S. Wu, G. Wang, P. Tang, F. Chen y L. Shi, «Convolution with even-sized kernels and symmetric padding,» en *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox y R. Garnett, eds., vol. 32, Curran Associates, Inc., 2019.
- [40] J. L. Ba, J. R. Kiros y G. E. Hinton, *Layer Normalization*, 2016.
- [41] L. Huang, J. Qin, Y. Zhou, F. Zhu, L. Liu y L. Shao, *Normalization Techniques in Training DNNs: Methodology, Analysis and Application*, 2020.

- [42] S. Ioffe y C. Szegedy, «Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,» en *Proceedings of the 32nd International Conference on Machine Learning*, F. Bach y D. Blei, eds., ép. Proceedings of Machine Learning Research, vol. 37, Lille, France: PMLR, jul. de 2015, págs. 448-456.
- [43] G. Huang, Z. Liu, L. van der Maaten y K. Q. Weinberger, «Densely Connected Convolutional Networks,» en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jul. de 2017.
- [44] M. Tan y Q. Le, «EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,» en *Proceedings of the 36th International Conference on Machine Learning*, K. Chaudhuri y R. Salakhutdinov, eds., ép. Proceedings of Machine Learning Research, vol. 97, PMLR, sep. de 2019, págs. 6105-6114.
- [45] F. Chollet, «Xception: Deep Learning With Depthwise Separable Convolutions,» en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jul. de 2017.
- [46] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie y L. Farhan, «Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions,» *Journal of big Data*, vol. 8, págs. 1-74, 2021.
- [47] T. Hoese y C. Kuenzer, «Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review-Part I: Evolution and Recent Trends,» *Remote Sensing*, vol. 12, n.º 10, 2020, ISSN: 2072-4292.
- [48] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong y Q. He, «A Comprehensive Survey on Transfer Learning,» *Proceedings of the IEEE*, vol. 109, n.º 1, págs. 43-76, 2021.
- [49] M. Raghu, C. Zhang, J. Kleinberg y S. Bengio, «Transfusion: Understanding Transfer Learning for Medical Imaging,» en *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox y R. Garnett, eds., vol. 32, Curran Associates, Inc., 2019.
- [50] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri y R. M. Summers, «Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases,» en *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, págs. 2097-2106.
- [51] Y. Peng, X. Wang, L. Lu, M. Bagheri, R. Summers y Z. Lu, «NegBio: a high-performance tool for negation and uncertainty detection in radiology reports,» *AMIA Summits on Translational Science Proceedings*, vol. 2018, pág. 188, 2018.
- [52] J. Bogatinovski, L. Todorovski, S. Džeroski y D. Kocev, «Comprehensive comparative study of multi-label classification methods,» *Expert Systems with Applications*, vol. 203, pág. 117215, 2022, ISSN: 0957-4174.

- [53] K. Sechidis, G. Tsoumakas e I. Vlahavas, «On the Stratification of Multi-label Data,» en *Machine Learning and Knowledge Discovery in Databases*, D. Gunopulos, T. Hofmann, D. Malerba y M. Vazirgiannis, eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, págs. 145-158.
- [54] P. Parsania, D. Virparia y col., «A review: Image interpolation techniques for image scaling,» *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 2, n.º 12, págs. 7409-7414, 2014.
- [55] C. Shorten y T. M. Khoshgoftaar, «A survey on image data augmentation for deep learning,» *Journal of big data*, vol. 6, n.º 1, págs. 1-48, 2019.
- [56] C. Khosla y B. S. Saini, «Enhancing Performance of Deep Learning Models with different Data Augmentation Techniques: A Survey,» en *2020 International Conference on Intelligent Engineering and Management (ICIEM)*, 2020, págs. 79-85.
- [57] J. P. Cohen, M. Hashir, R. Brooks y H. Bertrand, «On the limits of cross-domain generalization in automated X-ray prediction,» en *Medical Imaging with Deep Learning*, PMLR, 2020, págs. 136-155.
- [58] X. Zhang, X. Zhou, M. Lin y J. Sun, «ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices,» en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jun. de 2018.
- [59] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto y H. Adam, *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*, 2017.
- [60] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li y L. Fei-Fei, «ImageNet: A large-scale hierarchical image database,» en *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, págs. 248-255.
- [61] A. E. W. Johnson, T. J. Pollard, N. R. Greenbaum, M. P. Lungren, C.-y. Deng, Y. Peng, Z. Lu, R. G. Mark, S. J. Berkowitz y S. Horng, *MIMIC-CXR-JPG, a large publicly available database of labeled chest radiographs*, 2019.
- [62] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilicus, C. Chute, H. Marklund, B. Haghighi, R. Ball, K. Shpanskaya y col., «Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison,» en *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, 2019, págs. 590-597.
- [63] A. Bustos, A. Pertusa, J.-M. Salinas y M. de la Iglesia-Vayá, «PadChest: A large chest x-ray image dataset with multi-label annotated reports,» *Medical Image Analysis*, vol. 66, pág. 101 797, 2020, ISSN: 1361-8415.
- [64] M.-L. Zhang y Z.-H. Zhou, «A Review on Multi-Label Learning Algorithms,» *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, n.º 8, págs. 1819-1837, 2014.

-
- [65] W. Zhang, L. Deng, L. Zhang y D. Wu, «A Survey on Negative Transfer,» *IEEE/CAA Journal of Automatica Sinica*, vol. 10, n.º 2, págs. 305-329, 2022.
- [66] C. Sun, A. Shrivastava, S. Singh y A. Gupta, «Revisiting Unreasonable Effectiveness of Data in Deep Learning Era,» en *IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, págs. 843-852.
- [67] J. Hestness, S. Narang, N. Ardalani, G. Diamos, H. Jun, H. Kianinejad, M. Patwary, M. Ali, Y. Yang e Y. Zhou, «Deep learning scaling is predictable, empirically,» *arXiv preprint arXiv:1712.00409*, 2017.