UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

REDES NEURONALES DEL TIPO TRANSFORMER COMO HERRAMIENTA DE CLASIFICACIÓN DE SOBREEXPRESIÓN DE PROTEÍNA HER2 EN IMÁGENES DE BIOPSIAS DE CÁNCER GÁSTRICO

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL ELÉCTRICO

DIEGO IGNACIO MUÑOZ ROJAS

PROFESOR GUÍA:
CARLOS NAVARRO CLAVERÍA

MIEMBROS DE LA COMISIÓN:
MARCOS ORCHARD CONCHA
FRANCISCO RIVERA SERRANO

Este trabajo ha sido parcialmente financiado por
FONDECYT 1221696 y FONDEQUIP EQM210020.
Esta investigación fue apoyada por el supercomputador Patagón
de la Universidad Austral de Chile (FONDEQUIP EQM180042).

SANTIAGO DE CHILE
2023

RESUMEN DE LA MEMORIA PARA OPTAR
AL TÍTULO DE INGENIERO CIVIL ELÉCTRICO
POR: DIEGO IGNACIO MUÑOZ ROJAS
FECHA: 2023
PROF. GUÍA: CARLOS NAVARRO CLAVERIA

REDES NEURONALES DEL TIPO TRANSFORMER COMO HERRAMIENTA DE CLASIFICACIÓN DE SOBREEXPRESIÓN DE PROTEÍNA HER2 EN IMÁGENES DE BIOPSIAS DE CÁNCER GÁSTRICO

La presente memoria tiene como objetivo mejorar la precisión en la clasificación de imágenes de biopsias mediante el uso de redes neuronales de tipo *Transformer* en comparación con las redes convolucionales, para contribuir a mejorar el nivel de eficacia en el diagnóstico de cáncer gástrico. La motivación detrás de este proyecto radica en la capacidad de las redes neuronales de tipo *Transformer* para capturar relaciones contextuales complejas, lo que podría agilizar y mejorar el proceso de detección automática de características de interés en las biopsias. Un diagnóstico temprano y preciso del cáncer gástrico es fundamental para iniciar el tratamiento de manera oportuna, lo que puede mejorar significativamente las tasas de supervivencia y la calidad de vida de los pacientes.

El objetivo general del trabajo es aplicar y evaluar un modelo de redes neuronales de tipo *Transformer* en el proceso de detección automatizada de la sobre-expresión de la proteína HER2 en imágenes de biopsias de cáncer gástrico, con el fin de contribuir a un mayor nivel de eficacia en el diagnóstico de cáncer gástrico.

La metodología utilizada consistió en la separación de las biopsias en conjuntos de entrenamiento y prueba, así como el procesamiento de los datos para adaptarlos a la red neuronal. Se emplearon técnicas para aumentar la cantidad de datos, como la rotación y el reflexión de imágenes, para mejorar la diversidad y la cantidad de datos de entrenamiento. Se utilizó Python como lenguaje de programación y la librería PyTorch para implementar y entrenar las redes neuronales.

Para evaluar el rendimiento del modelo, se midió su capacidad para detectar la sobreexpresión de la proteína HER2 mediante la obtención de métricas de evaluación y la construcción de una matriz de confusión basada en las etiquetas y clasificaciones del modelo. Además, se realizó un análisis comparativo de los resultados obtenidos en este trabajo con estudios previos en el mismo campo de investigación, considerando aspectos como la precisión y la recuperación.

Finalmente, se ha comprobado que el *Transformer* presenta un mejor desempeño en comparación con la red convolucional (CNN) de referencia. El modelo desarrollado logra superar en 3 % el valor de exactitud de la CNN y, además, muestra un mejor *f1-score* en la clasificación de todas las clases.

A mi familia, por ser mi mayor motivación y apoyo incondicional en este camino.

Agradecimientos

Agradezco a mis padres, Ana y Luis, por haberme criado y por preocuparse siempre por mi educación. Especialmente, quiero expresar mi profundo agradecimiento a mi madre, quien ha sido mi principal apoyo a lo largo de mi vida y ha velado constantemente por mi bienestar emocional y mental.

También deseo agradecer a mi profesor guía, Carlos Navarro, quien siempre estuvo dispuesto a brindarme su ayuda y pacientemente me guió durante la realización de este trabajo.

Extendido mi agradecimiento a las personas que me han brindado su apoyo en este último período. A mis amigos Felipe Córdova, Miguel Videla, Alinson Lincopán, Diland Castro, Juan Faúndez y Luis Escares. En mi familia, quiero expresar mi gratitud a mi tía Patricia Muñoz y a mi hermano Álvaro Muñoz por su apoyo.

Tabla de Contenido

Índice de Tablas

Índice de Ilustraciones

Capítulo 1

Introducción

El cáncer es una de las causas de muerte más frecuentes en el mundo, debido a esto, es una enfermedad muy estudiada. La detección del cáncer y/o su tipo, es importante para realizar el tratamiento adecuado. En particular, el cáncer gástrico es el segundo tipo de cáncer más mortal de Chile.

Se comprobó que pacientes que padecen cáncer gástrico, con sobre-expresión de la proteína HER2, podrían beneficiarse del tratamiento con quimioterapia y el anticuerpo monoclonal Trastuzumab [1]. Para detectar la proteína HER2, las biopsias se someten a la acción de algunos anticuerpos para generar una respuesta particular, lo que provoca una coloración específica en las áreas donde la proteína HER2 está sobre-expresada.

Los patólogos detectan la sobre-expresión de la proteína HER2 en imágenes de biopsias en muy alta resolución, las cuáles pueden requerir una cantidad de tiempo sustancial para ser examinadas. El patólogo debe examinar la biopsia y clasificarla como 0 (Negativo), 1+ (Negativo), 2+ (equívoco) o 3+ (Positivo).

Debido al éxito que han tenido las redes neuronales en la clasificación de imágenes, surge la idea de crear una herramienta, que ayude a los patólogos en el análisis de imágenes de biopsias usando redes neuronales. La tecnología de redes neuronales proporciona una ventaja el análisis de biopsias tradicional, ya que es capaz de procesar grandes cantidades de datos de forma rápida y eficiente, lo cuál permitiría realizar un diagnóstico más eficiente y oportuno.

El objetivo principal de este estudio es realizar un análisis comparativo entre redes neuronales del tipo *Transformer* y trabajos previos realizados con redes convolucionales en la tarea de clasificación de imágenes de biopsias por trozos, con el propósito de mejorar y agilizar el diagnóstico de cáncer gástrico. Para lograr este objetivo, se utilizaron métricas de evaluación estándar que permiten medir la precisión y la recuperación de los modelos desarrollados.

Para alcanzar el objetivo establecido, se llevaron a cabo una serie de etapas. En primer lugar, fue necesario realizar una adaptación del formato de los datos para que estuvieran en un formato adecuado para la red neuronal. Luego, se procedió a realizar la separación de las biopsias en conjuntos de entrenamiento y evaluación utilizando el método de validación cruzada. Posteriormente, se procedió al entrenamiento de los modelos propuestos utilizando los conjuntos de datos previamente establecidos. Finalmente, se evaluaron los modelos utili-

zando las clasificaciones generadas por dichos modelos en los conjuntos de evaluación.

El reporte del trabajo de investigación realizado se secciona de la siguiente forma:

- Marco teórico: En este apartado se explican los elementos básicos relacionados con el proyecto, como las métricas para evaluar el rendimiento, las teorías subyacentes, los modelos utilizados y los conceptos clave relevantes.
- Estado del arte: se refiere a la revisión de la literatura actualizada y los trabajos previos relacionados con el tema de investigación.
- Implementación y evaluación: se describe detalladamente el enfoque y los procedimientos utilizados para llevar a cabo el estudio. Este apartado proporciona una visión de cómo se recopiló y analizó la información, así como los pasos específicos que se siguieron para abordar los objetivos de investigación.
- Resultados y Análisis: presenta y analiza los datos recopilados, interpretando y discutiendo los hallazgos en relación con los objetivos del estudio. Este apartado es esencial para comunicar los resultados de manera precisa y respaldar las conclusiones del informe.
- Conclusiones: resumen los principales hallazgos, extraen conclusiones basadas en los resultados y proporcionan recomendaciones para investigaciones futuras.

1.1. Objetivos

Objetivo general

Implementar y evaluar un modelo de redes neuronales del tipo *Transformer* en el proceso de detección automatizada de la sobre-expresión de la proteína HER2 en imágenes de biopsias de cáncer gástrico. El propósito de este enfoque es complementar la labor de los expertos en la detección y diagnóstico de cáncer gástrico, brindándoles una herramienta adicional que pueda mejorar su capacidad de toma de decisiones.

Objetivos específicos

- Procesar el *dataset* para obtener los datos de entrenamiento y evaluación del modelo de *Transformer*.
- Medir el rendimiento del modelo en la detección de la sobre-expresión de la proteína HER2, mediante la obtención de métricas de evaluación y la construcción de una matriz de confusión basada en las etiquetas y clasificaciones del modelo.
- Realizar un análisis comparativo de los resultados obtenidos en esta aplicación con trabajos previos en el mismo campo de investigación.

Capítulo 2

Marco Teórico y Estado del Arte

2.1. Marco Teórico

2.1.1. Biopsias

Las biopsias de cáncer gástrico son procedimientos médicos específicos realizados para obtener muestras de tejido del estómago de un paciente con el objetivo de diagnosticar y evaluar la presencia de cáncer en esta área.

El cáncer gástrico, también conocido como cáncer de estómago, es una enfermedad maligna que se origina en las células del revestimiento del estómago. Para confirmar o descartar la presencia de cáncer gástrico en un paciente, se lleva a cabo una biopsia. Durante este procedimiento, se toman pequeñas muestras de tejido del estómago utilizando instrumentos especiales, como una sonda endoscópica, que se inserta a través de la boca del paciente hasta alcanzar el estómago.

Una vez que se obtienen las muestras de biopsia se somete a una tinción inmunohistoquímica (IHC) la cual genera una coloración marrón la proteína HER2 está sobreexpresada. Luego las biopsias se envían a un laboratorio de patología donde se analizan bajo un microscopio. Los patólogos especializados en el estudio de tejidos examinan las muestras y determinan si hay presencia de células cancerosas, así como el tipo y grado de cáncer gástrico presente. Estos resultados son fundamentales para establecer un diagnóstico preciso y determinar el enfoque de tratamiento más adecuado para el paciente.

Para determinar el resultado de la biopsia los patólogos deben analizar la reactividad de la proteína HER2 en las células tumorales. Esto se hace determinando el porcentaje de células que tienen cierto tipo de reactividad, esto se muestra en la Figura 2.1.

Tabla 2.1: Tabla HER2 [2].

Espécimen quirúrgico	**Biopsia**	**Puntuación**	**Clasificación HER2**
No reactividad o reactividad membranosa en $< 10\,\%$ de las células tumorales	Sin reactividad en ninguna célula tumoral	IHC 0	Negativo
Reactividad membranosa débil o apenas perceptible en $\geq 10\,\%$ de las células tumorales; las células solo son reactivas en parte de su membrana	*Cluster* de células tumorales con reactividad membranosa débil o apenas perceptible, independientemente del porcentaje de células tumorales teñidas	IHC 1+	Negativo
Reactividad membranosa débil a moderada completa, basolateral o lateral en $\geq 10\,\%$ de las células tumorales	*Cluster* de células tumorales con reactividad membranosa débil a moderada completa, basolateral o lateral, independientemente del porcentaje de células tumorales teñidas	IHC 2+	Equivoca
Reactividad membranosa fuerte completa, basolateral o lateral en $\geq 10\,\%$ de las células tumorales	*Cluster* de células tumorales con reactividad membranosa completa, basolateral o lateral fuerte, independientemente del porcentaje de células tumorales teñidas	IHC 3+	Positivo

2.1.2. Métricas de evaluación

- **Matriz de confusión:** no es una métrica en sí misma, sino una visualización gráfica de la eficacia de un clasificador. Definiendo la matriz con la letra C cada elemento C_{ij} corresponde a la cantidad de elementos de la clase i que fueron clasificados como la clase j. La diagonal de la matriz contiene el número de elementos que se clasificaron correctamente. Esta información se utiliza como base para definir y explicar otras métricas. La Figura 2.1 muestra un esquema de una matriz de confusión junto con algunas métricas que se pueden deducir de ella.

- **Precisión:** corresponde a la proporción de los elementos clasificados como pertenecientes a la clase j que fueron correctamente clasificados. Esto se puede determinar a partir de la matriz de confusión utilizando la siguiente fórmula:

$$\text{Precisión }_j = \frac{C_{jj}}{\sum_{\text{i}=1}^{n} C_{\text{i}j}} \tag{2.1}$$

- **Recuperación (*recall*):** esta métrica determina cuántos elementos que son realmente de la clase i fueron identificados como tal por el clasificador. Esto se puede determinar a partir de la matriz de confusión utilizando la siguiente fórmula

$$\text{Recuperación }_{\text{i}} = \frac{C_{\text{ii}}}{\sum_{j=1}^{n} C_{\text{i}j}} \tag{2.2}$$

- **Exactitud (*accuracy*):** es una métrica global de desempeño de un clasificador. Se refiere a la cantidad de elementos clasificados correctamente, en relación con el número total de elementos clasificados.

$$\text{Exactitud } = \frac{\sum_{\text{i}=1}^{n} C_{\text{ii}}}{\sum_{\text{i}=1}^{n} \sum_{j=1}^{n} C_{\text{i}j}} \tag{2.3}$$

- ***F1-score*:** es una medida de rendimiento que se calcula como la media armónica entre la precisión y la recuperación de una clase dada. Esta técnica combina ambas métricas para obtener una medición de desempeño equilibrada.

$$\text{F1-score } = 2 \times \frac{\text{precisión } \times \text{ recuperación}}{\text{precisión } + \text{ recuperación}} \tag{2.4}$$

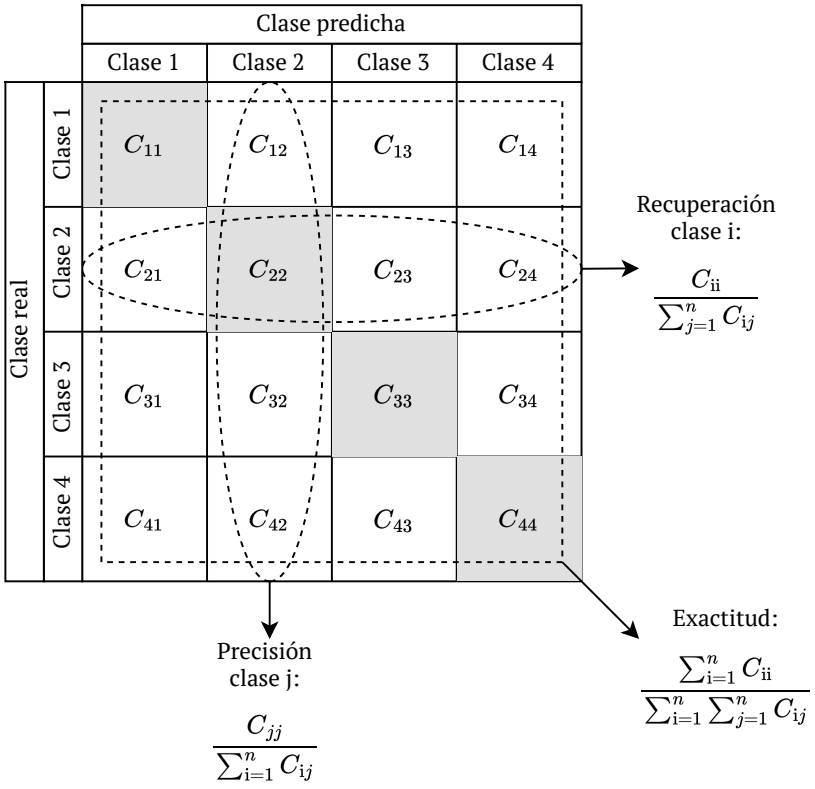


Figura 2.1: Esquema de matriz de confusión, junto a fórmulas derivadas.

2.1.3. *Transformer*

Transformer es un tipo de arquitectura de red neuronal diseñada inicialmente para el procesamiento de texto, y ha demostrado un rendimiento destacado en esta tarea. El concepto del *Transformer*, fue presentado en el *paper* "*Attention is all you need*" [3], y se basa en el uso de un mecanismo de atención que se aplica tanto en la etapa de codificación (*encoder*) como en la de decodificación (*decoder*).

La idea de *transformer* ha sido utilizada para generar otros modelos interesantes como BERT (*Bidirectional Encoder Representations from Transformers*), *SpanBERT*, *Transformer-XL* , *Compressive Transformer* y modelos GPT (*Generative Pre-trained Transformer*).

Aunque los *Transformer* se originaron en el procesamiento del lenguaje, su rendimiento excepcional ha llevado a su aplicación en diversos problemas, incluido el procesamiento de imágenes. Los *Transformers* se pueden utilizar para realizar múltiples tareas, como la traducción de texto, la generación de representaciones vectoriales para los *tokens* y la generación de texto basado en el contexto.

En resumen, los *Transformer* son una arquitectura de red neuronal versátil y poderosa que ha demostrado excelentes resultados en el procesamiento del lenguaje y, su aplicación se ha extendido a otras áreas, como el procesamiento de imágenes, brindando soluciones eficaces en diferentes tareas.

En esta sección se explicará el funcionamiento del transformer basándose en *Attention is all you need*[3], de donde se obtuvo la mayor parte de la información.

2.1.3.1. Arquitectura del *transformer*

El *Transformer* es una arquitectura compuesta por un *encoder* y un *decoder*. El *encoder* se encarga de procesar una secuencia de vectores de entrada $\{x_1, \ldots, x_n\}$ y transformarla en una secuencia de vectores $\{z_1, \ldots, z_n\}$. A partir de esta secuencia de vectores $\{z_1, \ldots, z_n\}$, el *decoder* genera otra secuencia de vectores $\{y_1, \ldots, y_m\}$, uno a la vez. Es importante destacar que para generar cada vector y_i, el *decoder* requiere la información de los vectores anteriores en la secuencia, ya que es un modelo auto-regresivo [3].

Encoder: está formado por N = 6 capas idénticas. Cada capa consta de dos subcapas. La primera es un mecanismo de *multi-head self-attention*, mientras que la segunda es una red MLP que procesa los vectores de manera independiente sin mezclarlos. Se utiliza una conexión residual alrededor de cada una de estas subcapas, seguida de una normalización de la capa. Esto implica que la salida de cada subcapa se calcula como Normalizacion(x + Subcapa(x)), donde Subcapa(x) es la función específica de la subcapa. Para facilitar estas conexiones residuales, todas las subcapas del modelo, así como las capas de *embeddings*, generan salidas de una dimensión $d_{model} = 512$.

Decoder: también está formado por N = 6 capas idénticas y cada capa está compuesta por 3 subcapas. La primera subcapa llamada *Masked multi-head attention* es un *multi-head self-attention*, pero el término *masked* se refiere a que no se utilizan todos los vectores y_i en su entrada sino sólo los que se han generado hasta el momento. La segunda subcapa es un *multi-head attention* que combina la salida del *encoder* con la subcapa anterior. La tercera subcapa es una red MLP. Al igual que en el caso del *encoder* aquí también se utiliza conexiones residuales y normalización.

Finalmente, para obtener la repuesta final de la red se aplica una proyección lineal y la operación *softmax* que entrega valores entre en el rango $[0, 1]$ (Ecuación 2.5). El diagrama de la Figura 2.2 resume la arquitectura del *Transformer* en [3].

$$\text{softmax}(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}} \quad \text{para } i = 1, \ldots, N \text{ y } \mathbf{z} = (z_1, \ldots, z_N) \in \mathbb{R}^N \tag{2.5}$$

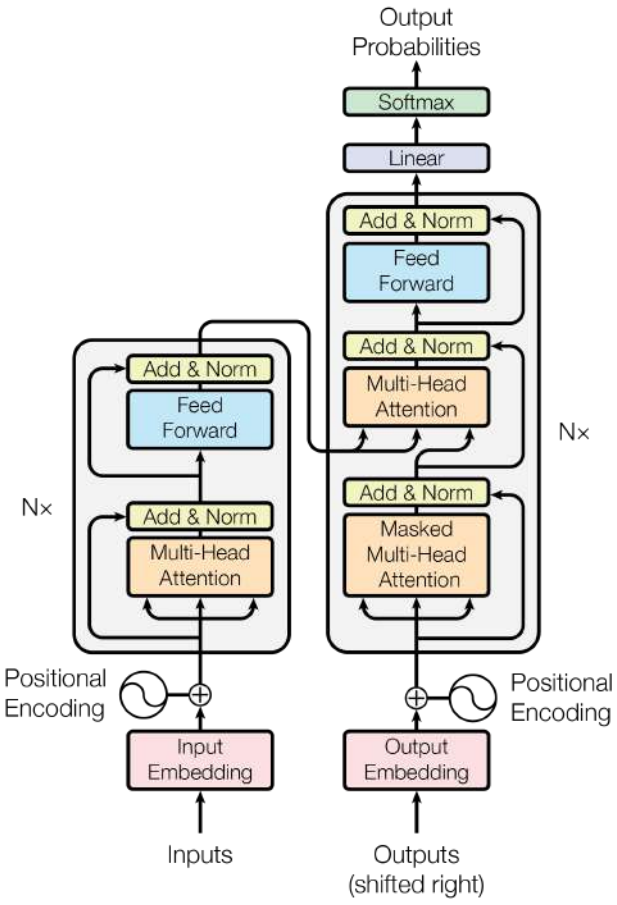


Figura 2.2: Arquitectura de Transformer. Obtenido de [3]

2.1.3.2. Mecanismo de *Attention*

El mecanismo de atención consiste en construir una representación $\{h_1, \ldots, h_m\}$ a partir de los vectores $V = \{v_1, \ldots, v_n\}$, conocidos como *values*, a través de una suma ponderada de todos ellos. La ponderación de cada vector se determina en función de su compatibilidad con otros vectores, la cual se calcula mediante vectores *keys* $K = \{k_1, \ldots, k_n\}$, asociados respectivamente a cada vector *value* y un vector *query* $Q = \{q_1, \ldots, q_m\}$ relacionado con el vector que se desea comparar.

Los vectores *query* y *key* se obtienen multiplicando vectores de una etapa anterior por una matriz[4]. Utilizando los parámetros entrenables W_q, W_k y W_v, se obtienen los siguientes vectores:

$$q_i\text{: vector } \textit{query} \text{ (consulta) de dimensión } d_k \qquad q_i = W_q \cdot y_i \tag{2.6}$$

$$k_j\text{: vector } \textit{key} \text{ (llave) de dimensión } d_k \qquad k_j = W_k \cdot x_j \tag{2.7}$$

$$v_j\text{: vector } \textit{value} \text{ (valor) de dimensión } d_v \qquad v_j = W_v \cdot x_j \tag{2.8}$$

Cuando los vectores x_j e y_i son idénticos, es decir, los vectores *query*, *key* y *value* se obtienen del mismo vector, se denomina *self-attention*.

La idea de atención es que para obtener el vector h_i se debe poner atención sobre un vector v_j. El vector *query* q_i representa a la consulta de sobre cuál de los vectores $V = \{v_1, \ldots, v_k\}$ se debe prestar atención. Para obtener esta respuesta se obtiene un *score* haciendo la multiplicación punto a punto de q_i con todos los vectores $K = \{k_1, \ldots, k_n\}$. Luego, a los puntajes obtenidos se le aplica la función *softmax* para obtener una distribución de probabilidad concentrada en el vector con mayor puntaje.

$$a_i = \text{Attention}(q_i, K, V) = \text{softmax}\left(\frac{q_i K^T}{\sqrt{d_k}}\right) V \tag{2.9}$$

La ecuación 2.9 muestra que el vector de atención a_i se obtiene a partir de una combinación lineal de los vectores $V = \{v_1, \ldots, v_n\}$, ponderados por los valores entregados por la función *softmax* que esta definida en la ecuación 2.5. Se divide por $\sqrt{d_k}$ porque cuando la dimensión d_k es grande el producto $q_i K^T$ entrega valores muy altos lo que lleva a la función *softmax* a regiones donde el gradiente es muy pequeño.

La operación de la ecuación 2.9 se realiza para todos los h_i en paralelo, por lo que en lugar de utilizar solo q_i se utiliza $Q = \{q_1, \ldots, q_m\}$ quedando la siguiente expresión:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{Q K^T}{\sqrt{d_k}}\right) V \tag{2.10}$$

En la Figura 2.3 se representan las operaciones de atención de la ecuación 2.10.

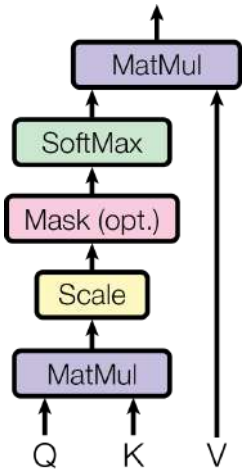


Figura 2.3: Operación de atención. Obtenido de [3].

2.1.3.3. *Multi-head attention*

El mecanismo de *Multi-head attention* realiza múltiples operaciones de atención en paralelo. Estas son conocidas como *heads* y utilizan vectores de *query*, *key* y *value* que han sido obtenidos mediante proyecciones lineales entrenables. Cada operación de atención y las proyecciones lineales asociadas tienen sus propios conjuntos de parámetros independientes. Una vez obtenidos los vectores de atención, estos se concatenan y, si es necesario, se les aplica otra proyección lineal entrenable para obtener un vector con las dimensiones requeridas. En la Figura 2.4 se muestra un diagrama que representa este proceso.

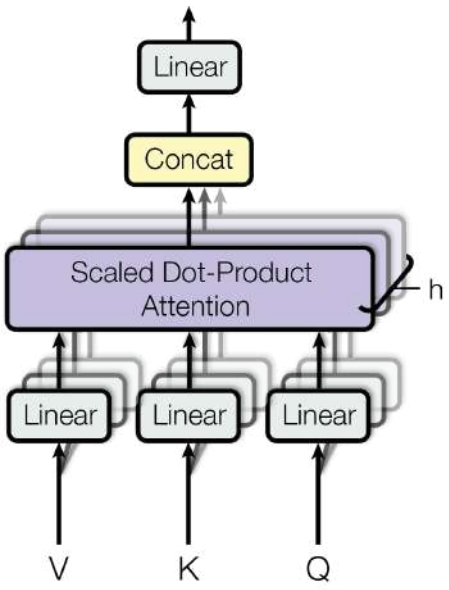


Figura 2.4: *Multi-head attention*. Obtenido de [3].

En la práctica, se pueden unir las proyecciones lineales del *Multi-head attention* con las usadas para obtener los vectores *query*, *key* y *value* (ecuaciones 2.6, 2.7 y 2.8) en una única proyección para cada *head*.

2.1.3.4. Distintas aplicaciones de atención en el modelo

- Atención en las capas del *encoder*: el *encoder* solo posee *self-attention* ya que todo los vectores se obtienen de la capas anteriores del *encoder*.
- Atención en las capas del *decoder*: la primera subcapa consiste en un *self-attention* que procesa la salida predicha anteriormente o la salida de la capa anterior. En la segunda subcapa de atención los vectores *key* y *value* se obtienen de la última capa del *encoder*, mientras que los vectores *query* se obtienen de la subcapa anterior del *decoder*. Esto se muestra en la Figura 2.2.

2.1.3.5. Codificación de la posición

Debido a la estructura del *transformer*, se pierde la información sobre el orden de la secuencia de vectores, lo cual es crucial para comprender el texto correctamente. Cambiar el orden de las palabras en una oración puede cambiar su significado. Para abordar este problema de las redes *transformer*, se utiliza un mecanismo que incorpora información de la posición absoluta o relativa de cada vector en relación con los demás. Existen diferentes opciones de vectores para representar la posición, por ejemplo, usar un parámetro entrenable, pero en [3] plantean los mostrados en la ecuación 2.11.

$$
\begin{aligned}
PE_{(pos,2i)} &= \sin\left(pos/10000^{2i/d_{\text{model}}}\right) \\
PE_{(pos,2i+1)} &= \cos\left(pos/10000^{2i/d_{\text{model}}}\right) \\
i \in \mathbb{Z} \wedge 0 \leq i &< \frac{d_{\text{model}}}{2}
\end{aligned}
\tag{2.11}
$$

Según la ecuación 2.11 en las posiciones pares se codifica con una función seno y en en las posiciones impares se codifica con una función coseno.

El vector de posición tiene la misma dimensión del *embedding* del input porque ambos son sumados para ser procesados por el *transformer*.

2.1.4. *Vision Transformer*

En 2020, el *Transformer* fue empleado para la tarea de Visión Computacional por primera vez en el artículo "An image is worth 16x16 words" [5]. El *Vision Transformer* ha tenido éxito en en distintas tareas como reconocimiento de imágenes, detección de objetos, segmentación, generación de imágenes, entre otras [6].

El *Vision Transformer* consiste en descomponer las imágenes en una serie de fragmentos llamados *patches*, los cuales, luego de ser transformados en vectores, son tratados de la misma forma que las palabras en un *Transformer encoder* corriente. Todas las capas de atención son del tipo *multihead self-attention* ya que sólo se utiliza el *encoder* del *Transformer*. En la Figura 2.5 se muestra un esquema sobre como se procesa la imagen en el *Vision Transformer*.

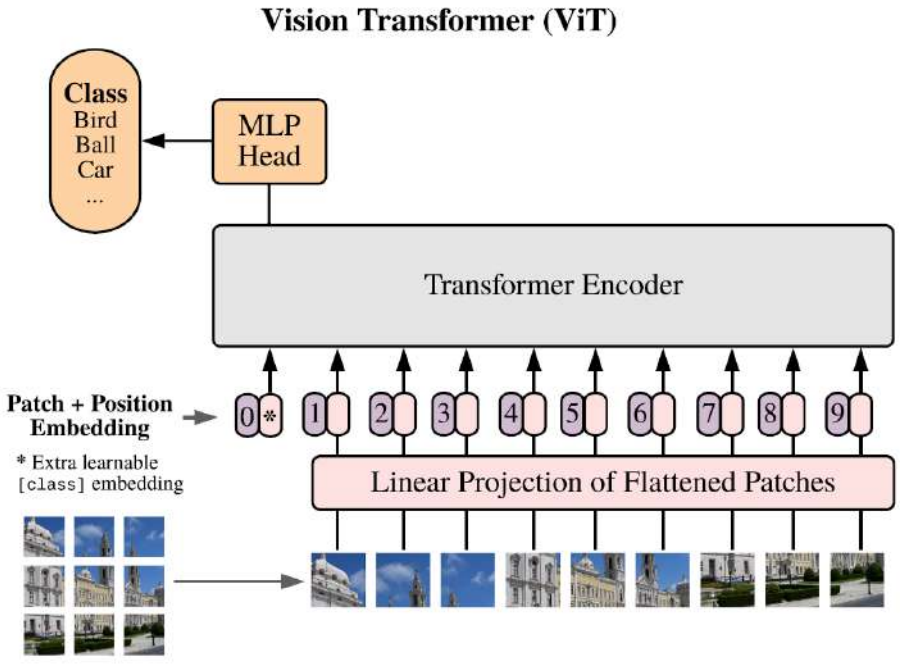


Figura 2.5: Esquema de arquitectura del *vision transformer*. Obtenido de [5].

Cuando se entrena con una cantidad de imágenes insuficiente, el *Transformer* puede presentar problemas de generalización en comparación con las redes convolucionales (CNN). Esto se debe a la falta de algunos sesgos inductivos inherentes a las CNN, como la equivalencia de traslación (la capacidad de reconocer patrones independientemente de su ubicación en la imagen) y la localidad (la capacidad de detectar patrones en regiones locales de la imagen). En resumen, los *Transformers* pueden tener un rendimiento deficiente en tareas de generalización si no se entrenan con una cantidad adecuada de datos [5].

2.1.4.1. Arquitectura

El *Transformer encoder* convencional está diseñado para procesar secuencias de *embeddings* de *tokens* en una dimensión (1D). Sin embargo, para aplicarlo a imágenes de 2 dimensiones (2D), es necesario transformar la imagen $\mathbf{x} \in \mathbb{R}^{H\times W\times C}$ en una secuencia de parches 2D apilados $\mathbf{x}_p \in \mathbb{R}^{N\times(P^2\cdot C)}$. Aquí, $(H;W)$ representa la resolución de la imagen original, C es el número de canales de la imagen (por ejemplo, 3 canales para una imagen RGB), y $(P;P)$ es la resolución de cada parche obtenido a partir de la imagen. La cantidad de parches resultantes se calcula como $N = HW/P^2$, lo que también corresponde a la cantidad de *tokens* que serán procesados por el *Transformer*[5].

Para mantener un tamaño constante D del vector latente en todas sus capas, los parches se reordenan en una secuencia y se transforman mediante una proyección lineal entrenable, representada por la matriz E en la ecuación 2.12. El resultado de esta proyección se conoce como los *embeddings* del parche y es el formato adecuado para ser procesado por el *Transformer*[5].

Se agrega un vector adicional ($\mathbf{z}_0^0 = \mathbf{x}_{class}$) en la entrada del *Transformer* para obtener un vector de salida adicional. En este vector de salida ($\mathbf{z}_L^0$) del *transformer encoder* se almacena y comprime los datos de la imagen para posteriormente ser procesado para realizar la clasificación final (ecuación 2.15). En el pre-entrenamiento, la clasificación la realiza una red MLP de una capa oculta y en *fine-tuning* la clasificación se realiza con una única capa lineal.

En [5], se realizaron experimentos con diferentes métodos de codificación de posición y se determinó que el uso de un *embedding* 1D entrenable es suficiente, ya que no se observó una mejora significativa en el rendimiento al utilizar métodos de codificación más complejos.

Las capas de *multihead self-attention* que conforman el *Transformer encoder* se representan mediante las ecuaciones 2.13 y 2.14, las cuales se repiten L veces. La ecuación 2.13 corresponde a la subcapa de *multihead self-attention* (MSA) junto con su normalización (LN) y conexión residual. Por otro lado, la ecuación 2.14 representa la subcapa de la red MLP, también con su normalización (LN) y conexión residual.

$$z_0 = \left[\mathbf{x}_{\text{class}}\,; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots ; \mathbf{x}_p^N\mathbf{E}\right] + \mathbf{E}_{\text{pos}}\,, \qquad \mathbf{E} \in \mathbb{R}^{\left(P^2\cdot C\right)\times D}, \mathbf{E}_{\text{pos}} \in \mathbb{R}^{(N+1)\times D} \tag{2.12}$$

$$z'_\ell = \text{MSA}\left(\text{LN}\left(\mathbf{z}_{\ell-1}\right)\right) + \mathbf{z}_{\ell-1}, \qquad \ell = 1\ldots L \tag{2.13}$$

$$\mathbf{z}_\ell = \text{MLP}\left(\text{LN}\left(\mathbf{z}'_\ell\right)\right) + \mathbf{z}'_\ell, \qquad \ell = 1\ldots L \tag{2.14}$$

$$\mathbf{y} = \text{LN}\left(\mathbf{z}_L^0\right) \tag{2.15}$$

2.1.5. Redes convolucionales

Las redes neuronales convolucionales, también conocidas como CNN (por sus siglas en inglés), son un tipo de arquitectura de redes neuronales profundas diseñadas específicamente para procesar datos en forma de matrices o tensores multidimensionales, como imágenes, por lo que estas redes son especialmente efectivas en tareas de clasificación, detección y reconocimiento de patrones en imágenes.

Una de las características clave de las redes convolucionales es su capacidad para aprender automáticamente características o filtros locales a través de capas de convolución. Estas capas convolucionales aplican filtros a pequeñas regiones de la imagen de entrada, conocidas como ventanas o *kernels*, y generan mapas de características convolucionales que resaltan patrones específicos, como bordes, texturas o formas, presentes en la imagen. En la Figura 2.6 se muestra como los *kernels* generan mapas de características.

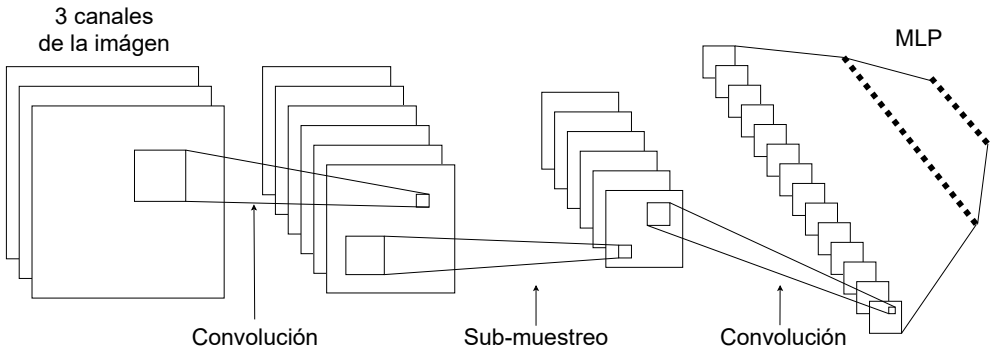


Figura 2.6: Arquitectura red convolucional.

Una vez aplicada la convolución, estas redes suelen incluir capas de *pooling*, también conocidas como capas de submuestreo. Estas capas reducen la dimensión espacial de los mapas de características y ayudan a extraer características más generales y robustas. Comúnmente, se utiliza la operación de *max pooling*, que selecciona el valor máximo en una ventana determinada, reduciendo así la resolución espacial.

Además de las capas convolucionales y de *pooling*, las redes convolucionales pueden incluir capas completamente conectadas o densas al final de la arquitectura. Estas capas finales se encargan de combinar y procesar las características extraídas previamente para realizar la tarea específica, como la clasificación de imágenes en categorías diferentes.

En resumen, las redes convolucionales son una arquitectura especializada de redes neuronales profundas que han demostrado un gran éxito en tareas de procesamiento de imágenes. Su capacidad para aprender características locales y su habilidad para capturar patrones visuales complejos las hacen especialmente adecuadas para aplicaciones de visión por computadora, como el reconocimiento de objetos, el análisis de imágenes médicas y el procesamiento de imágenes en general.

Las redes convolucionales poseen sesgos inductivos que corresponden a las suposiciones o prejuicios incorporados en el diseño y entrenamiento de estas redes. Estos sesgos influyen en

cómo la red aprende y representa los datos de entrada, lo que puede afectar la precisión y generalización de las predicciones realizadas por el modelo.

Los sesgos inductivos que poseen las redes convolucionales son:

- **Sesgo espacial**: las redes convolucionales están diseñadas para explotar la estructura espacial de los datos, asumiendo que las características y patrones relevantes se encuentran en regiones cercanas en el espacio. Esto es especialmente útil en imágenes, donde la proximidad espacial de píxeles o regiones puede contener información importante. Sin embargo, este sesgo puede resultar en una falta de capacidad para capturar patrones que se extienden a través de regiones más distantes.
- **Sesgo de traslación invariante**: las redes convolucionales están diseñadas para ser invariantes a las traslaciones en el espacio, lo que significa que deberían ser capaces de reconocer patrones similares independientemente de su ubicación en la imagen. Esto se logra utilizando operaciones de convolución y *pooling* que comparten pesos y reducen la dimensión espacial. Sin embargo, este sesgo puede llevar a que la red sea insensible a las variaciones finas de posición y a la orientación de los objetos en la imagen.

2.2. Estado del Arte

2.2.1. Clasificación automatizada de sobre expresión de proteína HER2 en biopsias digitalizadas de cáncer gástrico teñidas inmunohistoquímicamente

El objetivo principal de la tesis[7], es desarrollar un sistema informático capaz de realizar una clasificación automatizada de la sobreexpresión de la proteína HER2 en biopsias de cáncer gástrico que han sido digitalizadas. Para lograr esta clasificación automatizada, se trabajó con recortes de imágenes de las biopsias, conocidos como parches. Se empleó una red convolucional pre-entrenada llamada Inception-V3 [8].

Las biopsias utilizadas en este trabajo se obtuvieron del estudio *PRECISO* [2], cuyo objetivo es "Evaluar la eficacia y toxicidad de la quimioterapia perioperatoria con Epirubicina + Cisplatino + Capecitabina (ECX) en la práctica clínica habitual en una red de hospitales públicos de Santiago de Chile"[2]. Inicialmente, 61 personas se enrolaron en el estudio *PRECISO*, pero solo 48 autorizaron la determinación del nivel de sobreexpresión de la proteína HER2 a partir de sus muestras. Sin embargo, en la tesis, se utilizaron únicamente los datos de 40 de esos pacientes [7].

En el estudio *PRECISO*, originalmente, solo se disponía de la clasificación de la biopsia completa realizada por un solo patólogo, al que se le denominó patólogo 0. Sin embargo, para entrenar un modelo de clasificación, también fue necesario el etiquetado de regiones de interés (ROI) dentro de las imágenes de las biopsias para extraer recortes. Por lo tanto, se contó con la colaboración de dos patólogos, denominados patólogos 1 y 2, quienes demarcaron y etiquetaron las ROI en las biopsias, además de clasificar las biopsias completas [7].

2.2.1.1. Macro experimento I

En este trabajo, se llevó a cabo el primer macroexperimento utilizando las biopsias del estudio *PRECISO* y el etiquetado realizado por los patólogos 1 y 2; para el cual se extrajeron recortes de tamaño 300x300 píxeles a partir de cada Región de Interés (ROI) identificada en las biopsias. Estos recortes se obtuvieron utilizando magnificaciones de zoom de 10x, 20x y 40x. Se aplicó una superposición de 50 píxeles entre los recortes para asegurar una cobertura adecuada.

Posteriormente, se realizó un filtrado de los recortes para eliminar aquellos que presentaban menos del 20 % de tejido. De esta manera, se aseguró que los recortes seleccionados contuvieran una cantidad significativa de tejido relevante para el análisis.

En cuanto a las etiquetas utilizadas para entrenar el modelo, se empleó la clasificación HER2 mencionada en la Tabla 2.1. De este modo, se asignaron las siguientes etiquetas: negativo (IHC 0 y 1+), equívoco (IHC 2+) y positivo (IHC 3+). Se hicieron pruebas entrenando distintas cantidades de capas de la red; esto se muestra en la Tabla 2.2. Para tener más datos de entrenamiento, se crearon datos adicionales realizando distorsiones aleatorias como: rotaciones, desplazamientos, acercamiento, alejamiento y reflexiones.

El detalle de las distintas pruebas se muestra en la Tabla 2.2.

Tabla 2.2: Resumen de cada configuración experimental del macroexperimento I [7].

	Simple	*Data augmentation*	*Data augmentation + fine tuning*	Reentrenamiento total
Arquitectura red	Inception v3	Inception v3	Inception v3	Inception v3
Capas entrenables	Últimas 2 capas	Últimas 2 capas	Últimos 3 bloques de Inception y capas completamente conexas	Todas
Función de activación	Softmax	Softmax	Softmax	Softmax
Algoritmo de optimización	Adam	Adam	SGD con $\eta = 0,0001$	SGD con $\eta = 0,0001$
Función de costo	Entropía cruzada categórica	Entropía cruzada categórica	Entropía cruzada categórica	Entropía cruzada categórica
Tamaño batch	32	32	32	32
Distorsiones aleatorias	No	Sí	Sí	Sí

Alegría[7] concluyó que los resultados obtenidos en este macro experimento no fueron satisfactorios debido a la falta de concordancia entre las anotaciones de los diferentes patólogos.

2.2.1.2. Macro experimento II

Debido a las discrepancias encontradas entre los patólogos 1 y 2 en comparación con el patólogo 0, fue necesario obtener las anotaciones de otro patólogo, denominado patólogo 3. Este último patólogo generó anotaciones en 34 biopsias con el fin de mejorar la calidad y precisión de las clasificaciones [7]. En la Tabla 2.3 se muestra las etiquetas del patólogo 3 y la clasificación HER2 correspondiente a cada una.

En este macro experimento se utilizó 2 tipos de configuraciones ambas con *Inception-V3*.

- La primera, es un sólo modelo que clasifica directamente entre las 6 clases, es decir, clasifica entre no-tumor y los distintos niveles de reactividad.
- La segunda, consiste en 2 modelos que funcionan en cascada. Uno se encarga de clasificar entre tumor y no-tumor. Si el modelo deduce que es un tumor entonces el otro

Tabla 2.3: Anotaciones realizadas por patólogo 3 y clasificación HER2 correspondiente. Obtenido de [7]

Etiqueta Patólogo 3	Clasificación HER2 correspondiente
No tumor	No aplica
Sin reactividad	HER2 0
Reactividad positiva no lineal	HER2 0
Reactividad lineal casi imperceptible	HER2 1+
Reactividad lineal débil	HER2 2+
Reactividad lineal fuerte	HER2 3+

modelo clasifica el nivel de reactividad. En la Figura 2.7 se muestra un esquema de esta configuración.

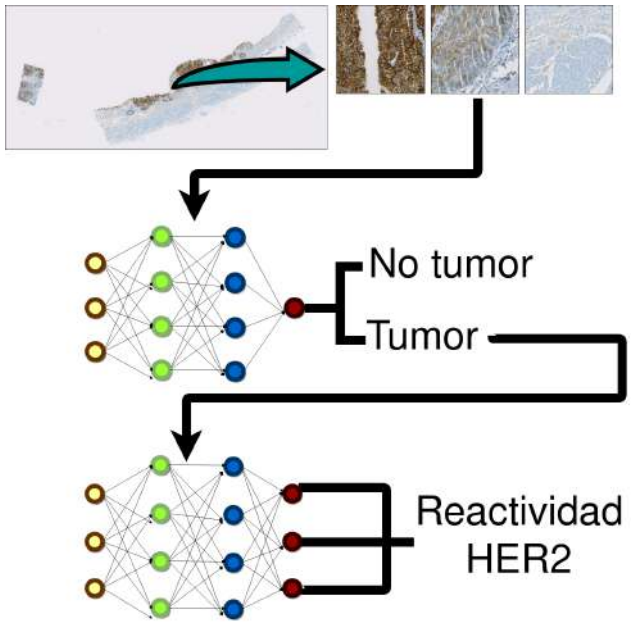


Figura 2.7: Esquema del experimento de modelos en cascada. Obtenido de [7]

En las Figuras 2.8 y 2.9 se muestran las matrices de confusión obtenidas para el experimento todo en uno y el experimento en cascada, respectivamente.

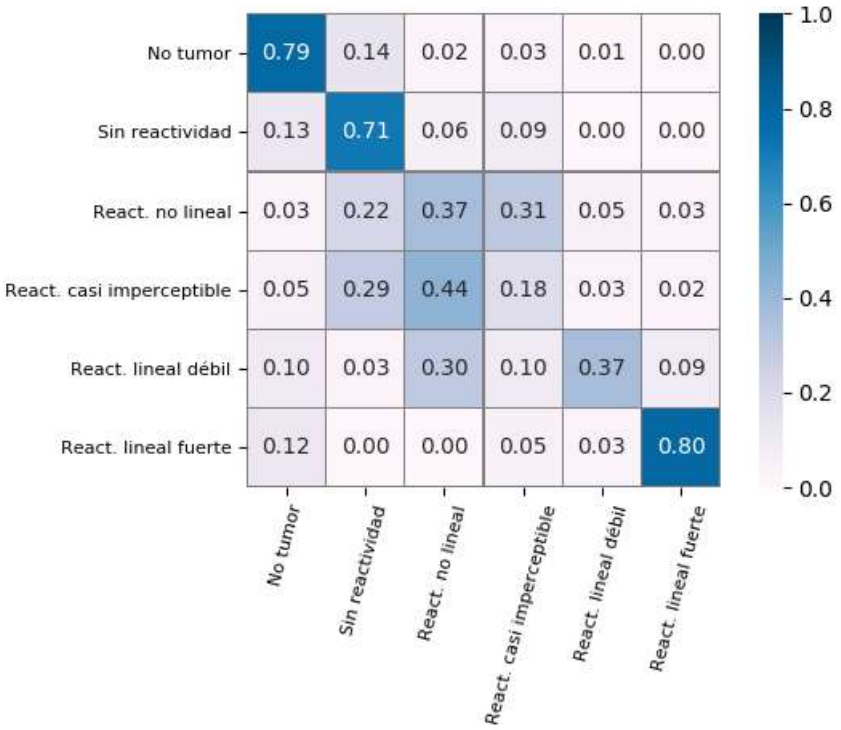
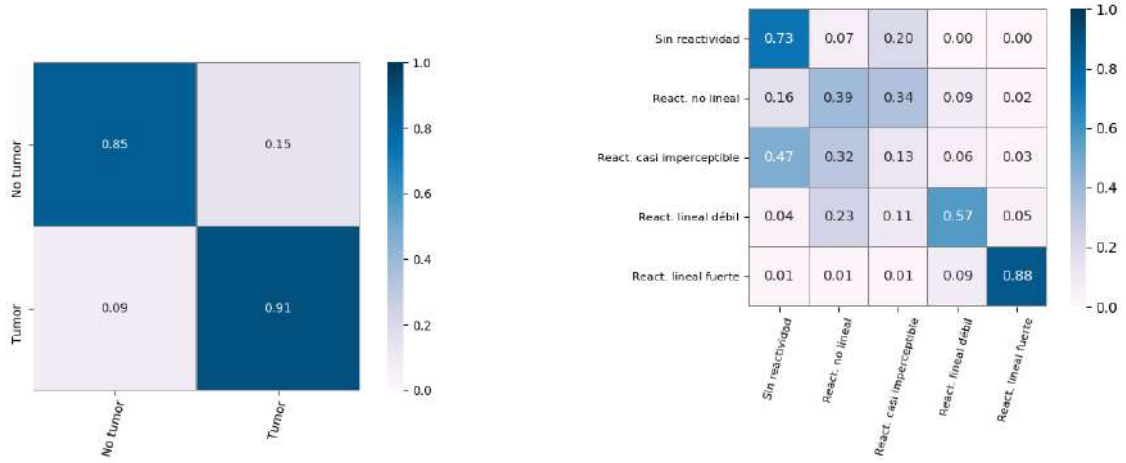


Figura 2.8: Matriz de confusión para experimento todo en uno [7]



(a) No tumor / Tumor

(b) Reactividad HER2

Figura 2.9: Matrices de confusión para experimento en cascada de [7]

En las Tablas 2.4 y 2.5 se muestran las métricas obtenidas para los modelos de tumor / no tumor y el de los 5 niveles de reactividad, respectivamente.

Tabla 2.4: Métricas de evaluación para modelo tumor y no tumor

	precisión	recuperación	f1-score	exactitud
NoTumor	0.92	0.85	0.89	-
Tumor	0.83	0.91	0.88	-
Promedio ponderado	0.88	0.88	0.88	0.88

Tabla 2.5: Métricas de evaluación para modelo de 5 niveles de reactividad

	precisión	recuperación	f1-score	exactitud
Sin reactividad	0.83	0.73	0.79	-
React. positiva no lineal	0.34	0.39	0.36	-
React. casi imperceptible	0.10	0.13	0.11	-
React. lineal débil	0.54	0.57	0.55	-
React. lineal fuerte	0.89	0.88	0.88	-
Promedio ponderado	0.65	0.61	0.63	0.61

En la Tabla 2.6 se muestra las métricas para el experimento todo en uno.

Tabla 2.6: Métricas de evaluación para modelo de 6 clases y con zoom x10 de Alegría[7]

	precisión	recuperación	f1-score	exactitud
No tumor	0.91	0.79	0.85	-
Sin reactividad	0.62	0.71	0.67	-
React. positiva no lineal	0.25	0.37	0.3	-
React. casi imperceptible	0.15	0.18	0.16	-
React. lineal débil	0.45	0.37	0.41	-
React. lineal fuerte	0.81	0.8	0.8	-
Promedio macro	-	-	-	0.7
Promedio ponderado	0.74	0.7	0.72	0.7

2.2.1.3. Síntesis de los Aspectos Relevantes

Se logró replicar el proceso de diagnóstico llevado a cabo por los médicos especialistas al clasificar la imagen completa, utilizando las reglas de los patólogos.

La Clasificación automatizada logró obtener resultados comparables a los entregados por los patólogos ya que la concordancia es comparable a la obtenida por ellos.

2.2.2. Clasificación de imágenes de cáncer gástrico aplicando aprendizaje profundo

En la memoria[9] se plantea un algoritmo para mejorar el rendimiento del modelo de Alegría[7]. El objetivo principal del algoritmo es filtrar el *dataset* original utilizado para entrenar la red de Alegría, con el fin de mantener solo los parches con etiquetado de mayor calidad. El algoritmo se compone de dos etapas principales: (1) Creación del filtro y (2) Aplicación del filtro.

2.2.2.1. Creación del filtro

En la etapa de creación del filtro, se estima un color característico o centroide para cada clase de reactividad HER2. Esto se logra mediante los siguientes pasos:

- Deconvolución de color: las imágenes se transforman en el espacio de colores H-DAB para simplificar la segmentación de núcleos celulares y membranas.

- Segmentación basada en umbral de Otsu: se utiliza el algoritmo de Otsu para segmentar las células del fondo de la imagen.

- Identificación de región celular: se identifican las regiones de donde se extraen los colores H-DAB. Se generan ventanas celulares alrededor del centro de cada célula para extraer los colores.

- Estimación de centroides de colores: se obtiene un color representativo asociado a cada clase de reactividad HER2 mediante la agrupación de ventanas celulares asociadas a parches de una cierta clase. Estos centroides son puntos en el espacio H-DAB.

2.2.2.2. Aplicación del filtro

En la etapa de aplicación del filtro, se realiza el filtrado de parches utilizando los centroides de colores obtenidos en la etapa anterior. El proceso se lleva a cabo en cada parche del *dataset* PRECISO y se compone de los siguientes pasos:

- Cálculo de color representativo del parche: se calcula un color representativo para cada parche mediante el promedio de los colores H-DAB asociados a las células detectadas en el parche.

- Regla de decisión de filtro: se calculan las distancias entre el color representativo del parche y los centroides de cada clase de reactividad HER2. Si la clase del parche coincide con la clase asociada a la distancia más corta, el parche se mantiene en el *dataset* filtrado; de lo contrario, se elimina.

Al finalizar el proceso, se obtiene un *dataset* filtrado compuesto por parches etiquetados con mayor calidad, y se espera que reentrenando los modelos con este *dataset* mejore su

rendimiento.

2.2.2.3. Síntesis de los Aspectos Relevantes

En resumen, el algoritmo propuesto consta de dos etapas: la primera etapa se encarga de estimar los centroides de colores representativos para cada clase, mientras que la segunda etapa aplica el filtro para mantener solo los parches con etiquetado de mayor calidad, utilizando las distancias entre los colores representativos y los centroides.

El filtrado del *dataset* se aplicó para entrenar modelos clasificación de tumor / no tumor y de clasificación de los 5 niveles de reactividad de la proteína HER2. En la Tabla 2.7 se muestran las métricas obtenidas.

Tabla 2.7: Métricas resultados de clasificación del modelo Tumor/No tumor y del modelo de 5 Clases Reactividad

	Precisión	Recuperación	F1-score	Exactitud
Tumor/No tumor	0.79	0.78	0.79	0.86
5 Clases Reactividad	0.46	0.34	0.39	0.504

Según las Tablas 2.7 y 2.4, se puede observar que el modelo de clasificación de Tumor/No tumor propuesto por Escares[9] obtuvo un desempeño ligeramente inferior en términos de exactitud en comparación con el modelo de Alegría[7]. Sin embargo, en el resto de las métricas evaluadas, se obtuvieron considerablemente peores resultados para el modelo de Escares[9].

Las Tablas 2.7 y 2.5 muestran que el modelo para la clasificación de los 5 niveles de reactividad de Escares[9] obtuvo claramente peores resultados que el modelo de Alegría[7].

El algoritmo de filtrado de imágenes no parece ser efectivo para mejorar la calidad de los datos de entrenamiento. Es importante destacar que el proceso de filtrado puede eliminar información relevante y limitar la diversidad del *dataset*, lo que podría tener un impacto negativo en el desempeño del modelo.

Capítulo 3

Implementación y Evaluación del Modelo de clasificación de sobreexpresión de proteína HER2

En este capítulo se explica las técnicas usadas para la resolución del problema y cómo se evaluaron los resultados.

La metodología de este trabajo se fundamenta en gran medida en [7], ya que se emplea el mismo conjunto de datos para llevar a cabo la investigación. Esto es crucial, ya que brinda la oportunidad de realizar una comparación directa con los resultados obtenidos en [7].

El *dataset* está compuesto por 34 archivos *ndpi* y *ndpa* pertenecientes al estudio *PRECISO*[2]. Los archivos *ndpi* contienen representaciones de la imagen en esquema piramidal con distintos niveles de magnificación. La imagen con mayor resolución es la de *zoom* x40 y, a partir de ella se generan otras versiones con menor resolución, donde en cada nivel se disminuye la resolución a la mitad. Por lo tanto, los niveles disponibles son x40, x20, x10, x5 y x2.5. Cada archivo *ndpa* corresponde a las anotaciones de las imágenes *ndpi*.

En este proyecto se utilizó el lenguaje de programación Python y algunas de sus librerías como pandas 1.3.4, numpy 1.24.3, openslide 1.2.0, torch 1.10.0 y torchvision 0.11.1. La librería openslide se utiliza para obtener imágenes e información de los archivos *ndpi*.

3.1. Pre-procesamiento de los datos

Las anotaciones consisten en círculos que delimitan el tejido que corresponde a una cierta etiqueta. Los patólogos utilizan diferentes colores para denotar diferentes niveles de reactividad de la proteína HER2. En la Figura 3.1 se muestra un ejemplo de los círculos que corresponden a las anotaciones de los patólogos.

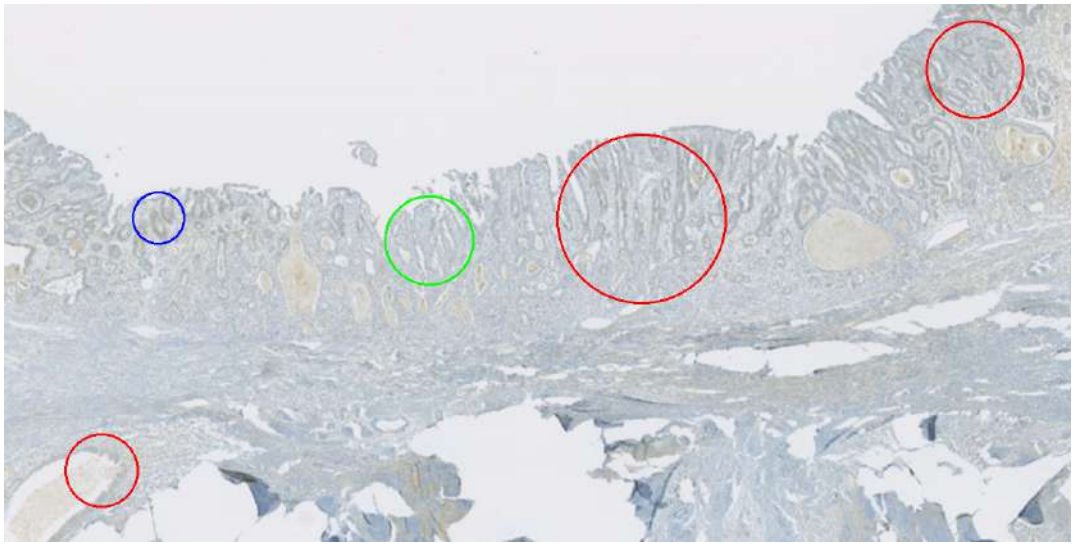


Figura 3.1: Ejemplo de anotaciones del patólogo

3.1.1. Selección de parches

El modelo de clasificación de imágenes utiliza un tamaño fijo de imagen, por lo que se requiere segmentar la imagen de la biopsia en múltiples secciones para que el modelo pueda realizar la clasificación adecuada. Estas secciones deben ser recortados según las anotaciones realizadas por el patólogo 3 y etiquetados de manera precisa para que el modelo pueda aprender a detectar y clasificar las características específicas de cada tipo de célula o tejido presentes en la imagen.

El modelo usado requiere imágenes de 224x224 píxeles. Para realizar los recortes, se crea una grilla de cuadrados de 224x224 píxeles que cubren los círculos. Solo se utilizan aquellos cuadrados que contienen al menos el 70 % de su área dentro del círculo. En la Figura 3.2 se muestran ejemplos de cómo se lleva a cabo este proceso, donde los números dentro de los cuadrados indican la proporción de área del cuadrado que queda dentro del círculo.

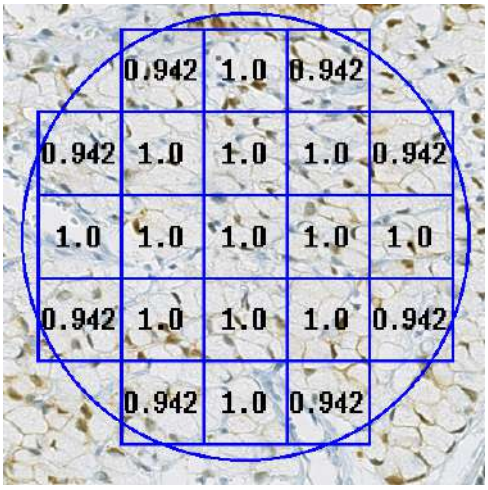


Figura 3.2: Recorte de parches

Luego de la segmentación de la imagen en sub-imágenes de 224x224, se crea un sistema de detección de segmentos no relevantes para la clasificación. Como criterio se utilizó la proporción de píxeles en blanco en dicha muestra, esto se debe a que píxeles en blanco están altamente ligados al fondo de la muestra y, por tanto, acarrean poca información estadística con respecto al problema de clasificación planteado. En la Figura 3.3, se pueden apreciar los cuadrados de color celeste que representan los parches eliminados debido a que menos del 20 % de sus píxeles se consideraban tejido según este criterio. Esta técnica se basa en el enfoque presentado en [7].

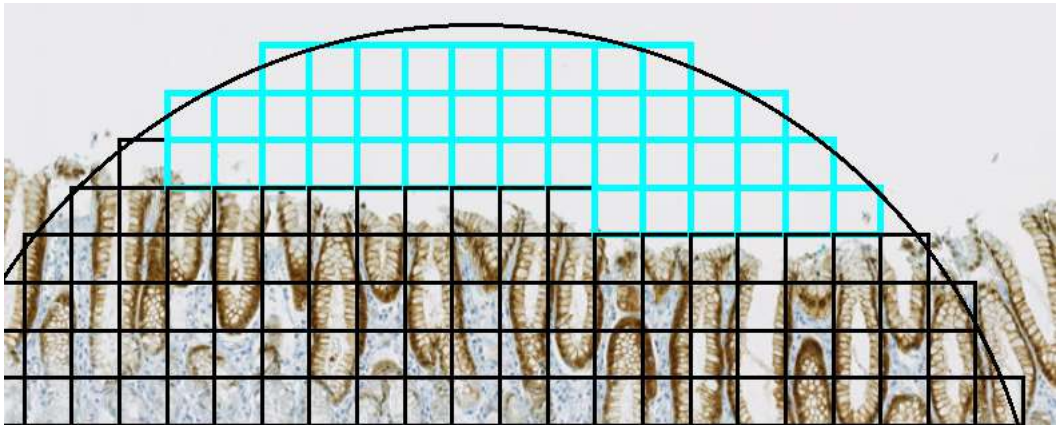


Figura 3.3: Área sin tejido

3.1.2. Recorte de parches

En el proceso de generación de recortes de imágenes, se les asigna a cada uno la etiqueta correspondiente a la región de interés (ROI) de la cual se originan. Los recortes se obtienen de los niveles de *zoom* x10, x20 y x40.

Con el objetivo de incorporar rotaciones en el proceso de entrenamiento, se realiza una ligera ampliación en el tamaño de los recortes. Esto se hace para evitar el impacto de secciones vacías en la imagen luego de aplicar rotaciones. El tamaño del recorte utilizado fue de 317 píxeles y se calculó de la siguiente forma: tamaño del recorte $= \lceil (224 \cdot \sqrt{2}) \rceil$, considerando que el modelo necesita imágenes de 224x224 píxeles. Este es el máximo tamaño que puede ser requerido, lo cual ocurre con una rotación de 45º, 135º, -45º y -135º.

Es importante tener en cuenta que, debido al margen extra necesario para la aplicación de rotaciones, se genera cierta superposición entre los recortes. Esta superposición se produce como resultado del espacio adicional requerido para acomodar las rotaciones dentro de cada recorte. Aunque exista cierta superposición entre los recortes, esta característica contribuye a enriquecer el conjunto de datos de entrenamiento al proporcionar mayor variabilidad y promover una mejor generalización del modelo.

En el nombre del archivo del recorte, se almacenó *metadata*, como el nombre del archivo de biopsia del que proviene, el número identificador de la anotación y, también, se guardó la información utilizada en el comando de extracción del recorte desde el archivo de la biopsia. Esto incluye: la coordenada de la posición de extracción, el tamaño del recorte y el nivel de

zoom desde el que se extrajo. Estos datos son necesarios para posteriormente verificar que los recortes se efectuaron correctamente.

3.1.3. Aplicación de visualización de recortes

Se desarrolló una aplicación para comprobar el correcto funcionamiento de los recortes, la cual permite observar cómo se distribuyen los parches dentro de las anotaciones de las biopsias. Para ello, se diseñó un sistema que utiliza las representaciones de esquema piramidal de los archivos *ndpi*, ajustando dinámicamente la resolución de la imagen mostrada de acuerdo al acercamiento requerido por el usuario. De esta manera, el usuario puede examinar detalladamente las anotaciones y parches, garantizando la precisión y calidad de los recortes generados.

- Muestra los círculos que corresponden a las anotaciones de los patólogos. Un ejemplo de esto se muestra en la Figura 3.1.

- Muestra como se reparten los parches dentro de las anotaciones de los patólogos y la porción de área que queda dentro del círculo. Un ejemplo de esto se muestra en la Figura 3.2.

- Muestra con un color distinto los parches descartados por tener muy poco tejido. Un ejemplo de esto se muestra en la Figura 3.3.

- A partir de la información contenida en el nombre de un archivo de recorte, la aplicación muestra la zona de la biopsia de donde proviene, marcando el rectángulo que corresponde a este recorte. Además, se muestra la anotación del patólogo asociada, junto con sus parches, para que el usuario pueda comprender el contexto del recorte y verificar que todo funciona correctamente.

- Muestra cómo los modelos realizaron la clasificación de los parches mediante la asignación de colores a cada cuadrado según su etiqueta.

3.2. Entrenamiento del modelo

3.2.1. Asignación de conjuntos de entrenamiento

Existe un desbalance en la cantidad de recortes de cada clase y a la mayoría de las biopsias les faltan anotaciones de algunas más de una clase. Debido a estas razones, se hace difícil aplicar una estrategia de validación cruzada *K-Fold* como se menciona en [7]. Por lo tanto, al igual que en dicho trabajo, se ha decidido evaluar el desempeño utilizando la estrategia de validación cruzada *leave-one-out cross-validation[10] (LOOCV)* tanto para abordar el desbalance como para permitir una comparación adecuada.

La estrategia de *LOOCV* fue aplicada a nivel de las biopsias en lugar de los recortes, siguiendo el enfoque utilizado en [7]. Esta elección se debe a que los recortes provenientes de una misma biopsia pueden presentar correlación entre sí, lo que puede afectar la validez de la estrategia de validación cruzada.

Aplicar la estrategia de *LOOCV* a las 34 biopsias implica entrenar 34 modelos distintos, donde cada modelo se entrena con 33 biopsias y se reserva una biopsia diferente para *test* en cada modelo.

3.2.2. Modelo pre-entrenado

Se utilizó el modelo pre-entrenado [11]; un modelo que fue entrenado con el *dataset ImageNet-21k*. El modelo que se está utilizando es una red neuronal del tipo *vision transformer* construida en el *framework* de *deep learning* llamado PyTorch. Sin embargo, se ha utilizado la biblioteca externa *timm* para cargar y utilizar un modelo preentrenado. La librería *timm* es una biblioteca de modelos de visión por computadora que proporciona una amplia variedad de arquitecturas de redes neuronales preentrenadas.

3.2.3. Entrenamiento

En este estudio, se llevaron a cabo entrenamientos de modelos de clasificación para determinar la reactividad de la proteína HER2 junto con las células etiquetadas como no tumor, abarcando un total de 6 clases según se describe en la Tabla 2.3. Cada entrenamiento se realizó durante 30 épocas y se utilizó la estrategia de leave-one-out cross-validation (LOOCV).

Para iniciar los entrenamientos, se emplearon los parámetros pre-entrenados del modelo[11]. Durante todos los entrenamientos, se utilizó la función de costo *weighted cross entropy*, la cual es ampliamente reconocida por su efectividad en tareas de clasificación. El optimizador empleado fue el descenso estocástico del gradiente con un *learning rate* de 10^{-3}.

En línea con [7], donde también se enfrentó desbalance de clases en los datos, se incorporó un enfoque para la función objetivo de la red neuronal. En esta aproximación, se asignaron pesos inversamente proporcionales a la cantidad de datos de cada clase, con el objetivo de equilibrar la influencia de las clases minoritarias en el entrenamiento y mejorar el rendimiento general del modelo.

El tamaño de batch elegido fue de 45, ligeramente por debajo de la capacidad total de memoria de la GPU. Esta decisión se tomó para asegurar que el entrenamiento se realice de manera eficiente y que la GPU pueda manejar el procesamiento de los datos sin sobrecargarse.

3.2.4. Escalado de las imágenes y *Data augmentation*

Se empleó la técnicas de *data augmentation*, con el propósito de generar datos artificiales durante el entrenamiento. Esta estrategia tiene como objetivo evitar el sobre-ajuste del modelo al introducir mayor variabilidad en los datos de entrenamiento. Al aplicar transformaciones a los datos existentes, se amplía la diversidad de ejemplos disponibles, permitiendo que el modelo aprenda características más robustas y generalizables.

A diferencia de las fotos comunes, las biopsias no poseen una orientación fija. Esto permite utilizar rotaciones aleatorias en cualquier ángulo, es decir, entre -180º y 180 grados. Después de la rotación, se extrae la imagen del centro con un tamaño de 224x224 píxeles. Es importante mencionar que esta extracción se realiza justo después de la rotación para evitar realizar operaciones innecesarias en píxeles que no se usarán.

También, se aplicó un reflexión horizontal con una probabilidad del 50 %. No se utilizó una reflexión vertical, ya que sería redundante en combinación con la rotación y el reflexión horizontal. Finalmente, se realiza un escalado y una traslación de los valores de la imagen para que queden en el rango [-1, 1]. La ecuación 3.1 muestra cómo se obtiene el nuevo valor de los píxeles P_n a partir de los valores originales de los píxeles P_o.

$$P_n = \frac{\frac{P_o}{255} - 0.5}{0.5} \tag{3.1}$$

En resumen las operaciones sobre los recortes durante el entrenamiento quedan de en el siguiente orden:

- Rotación aleatoria entre -180 y 180 grados.
- Extracción de la imagen del centro con un tamaño de 224x224 píxeles.
- Reflexión horizontal con probabilidad del 50 %.
- Escalado y traslación de los valores de la imagen según la ecuación 3.1.

3.3. Experimento con niveles de *zoom*

Se llevaron a cabo entrenamientos utilizando imágenes de tres niveles de *zoom*: x10, x20 y x40, con el propósito de determinar cuál sería el más adecuado para el modelo. No se exploraron otros niveles de *zoom* debido a que aquellos menores a x10 proporcionarían un número muy reducido de imágenes para el entrenamiento. En la Tabla 3.1 se muestra la cantidad de recortes obtenidos para cada clase y para cada nivel de *zoom*.

Tabla 3.1: Cantidad de recortes obtenidos en cada nivel de *zoom*

	Zoom x10	Zoom x20	Zoom x40
No tumor	2370	9467	37402
Sin reactividad	1137	4613	18484
React. positiva no lineal	295	1201	4875
React. casi imperceptible	321	1386	5640
React. lineal débil	148	655	2639
React. lineal fuerte	184	766	3099
Total	4455	18088	72139

En el caso del entrenamiento con imágenes x40, se enfrentó el desafío de contar con una cantidad considerable de imágenes, lo que podría prolongar en exceso el entrenamiento de los 34 modelos. Considerando el exceso de imágenes disponibles en las clases "no tumor" y "sin reactividad", se optó por utilizar una cantidad reducida de imágenes de estas clases en cada época de entrenamiento. Antes de cada época, se seleccionaron aleatoriamente un subconjunto de imágenes. Específicamente, se utilizó la octava parte de los recortes de la clase "no tumor" y la cuarta parte de los recortes de la clase "sin reactividad".

3.4. Experimento con modelos de clasificación en cascada

Cuando un patólogo evalúa una biopsia de resección, debe considerar cuidadosamente el porcentaje de células tumorales que presentan cierto tipo de reactividad, tal como se ilustra en la Tabla 2.1. Debido a esta consideración, al momento de evaluar la biopsia, no se debe no tomar en cuenta las células no tumorales. Dado que en las biopsias, en general, la mayoría del tejido no es tumor, surge la oportunidad de explorar un enfoque distinto.

La propuesta de utilizar un modelo en cascada es interesante, ya que podría ofrecer ventajas significativas en términos de eficiencia y precisión. Al emplear primero un modelo para identificar el tumor, este procesaría todo el tejido de la biopsia, lo que permitiría considerar la elección de un modelo más rápido y liviano que pueda procesar las imágenes con mayor celeridad. Por otro lado, el segundo modelo, al enfocarse únicamente en la clasificación de la reactividad, trabajaría con un conjunto reducido de imágenes, lo que brindaría la oportunidad de optar por un modelo más complejo y preciso, aunque pueda ser más lento en su ejecución.

Se entrenaron 2 modelos utilizando nivel de *zoom* que entregó mejores resultados en el experimento anterior:

- Modelo de clasificación Tumor /no Tumor: para entrenar este modelo se agruparon todas las etiquetas de tipo de reactividad en la etiqueta Tumor.
- Modelo de clasificación de 5 tipos de reactividad: para entrenar este modelo no se usaron los recortes con la etiqueta no Tumor.

Capítulo 4

Resultados y discusión

A continuación se muestran los resultados obtenidos utilizando métricas de clasificación, que incluyen matriz de confusión, precisión, recuperación, exactitud y *f1-score*.

4.1. Selección de nivel de *zoom*

A continuación se muestran las matrices de confusión normalizadas obtenidas al concatenar los resultados de la clasificación de los 34 modelos sobre sus respectivos archivos de prueba. Estos modelos fueron entrenados con imágenes obtenidas en niveles de *zoom* x10, x20 y x40. Las Figuras 4.1, 4.2 y 4.3 corresponden a las matrices de confusión de cada nivel de *zoom*, respectivamente.

Figura 4.1: Matriz de confusión clasificación 6 clases con *zoom* x10

Figura 4.2: Matriz de confusión clasificación 6 clases con *zoom* x20

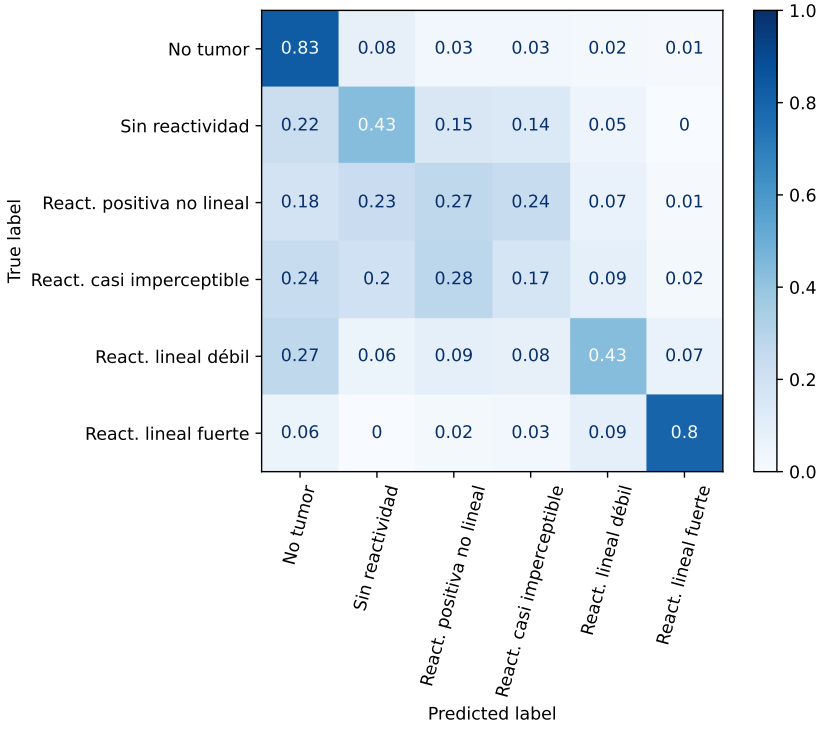


Figura 4.3: Matriz de confusión clasificación 6 clases con *zoom* x40

Las Tablas 4.1, 4.2 y 4.3 muestran las métricas obtenidas en la clasificación de las 6 clases con distintos niveles de *zoom*. Están destacados los mejores valores obtenidos al comparar los 3 casos.

Tabla 4.1: Métricas de evaluación para 6 clases y con zoom x10

	precisión	recuperación	f1-score	exactitud
No tumor	**0.91**	**0.9**	**0.91**	-
Sin reactividad	**0.71**	0.63	**0.66**	-
React. positiva no lineal	**0.31**	0.26	**0.29**	-
React. casi imperceptible	0.13	**0.19**	0.15	-
React. lineal débil	**0.48**	**0.55**	**0.51**	-
React. lineal fuerte	0.8	**0.92**	**0.86**	-
Promedio macro	**0.56**	**0.58**	**0.56**	**0.73**
Promedio ponderado	**0.74**	**0.73**	**0.73**	**0.73**

Tabla 4.2: Métricas de evaluación para 6 clases y con zoom x20

	precisión	recuperación	f1-score	exactitud
No tumor	0.87	0.88	0.88	-
Sin reactividad	0.67	**0.64**	0.65	-
React. positiva no lineal	0.21	**0.29**	0.24	-
React. casi imperceptible	0.13	0.08	0.1	-
React. lineal débil	0.34	0.42	0.37	-
React. lineal fuerte	**0.85**	0.86	0.85	-
Promedio macro	0.51	0.53	0.52	0.7
Promedio ponderado	0.7	0.7	0.7	0.7

Tabla 4.3: Métricas de evaluación para 6 clases y con zoom x40

	precisión	recuperación	f1-score	exactitud
No tumor	0.81	0.83	0.82	-
Sin reactividad	0.6	0.43	0.5	-
React. positiva no lineal	0.18	0.27	0.22	-
React. casi imperceptible	**0.15**	0.17	**0.16**	-
React. lineal débil	0.29	0.43	0.34	-
React. lineal fuerte	0.77	0.8	0.79	-
Promedio macro	0.47	0.49	0.47	0.62
Promedio ponderado	0.64	0.62	0.63	0.62

Se observa, según la Tabla 4.1, que con imágenes con *zoom* x10 se obtuvieron los mejores valores para la mayoría de las métricas. Es especialmente relevante destacar los valores de recuperación para las clases “react. lineal débil” y “react. lineal fuerte”, ya que identificar correctamente estas clases permitiría aplicar un tratamiento adecuado para mejorar la su-

pervivencia de los pacientes. Utilizando las imágenes con *zoom* x10 se lograron los mejores valores de recuperación para estas clases.

Cabe destacar que al cambiar el nivel de *zoom*, también se modifica el tipo de información disponible. Un nivel *zoom* mayor permite obtener información de una superficie más pequeña, pero con mayor detalle. Por otro lado, un *zoom* menor abarca un área más amplia pero con menos detalle. Esta característica puede influir en el rendimiento del modelo al procesar las imágenes, ya que la selección del nivel de *zoom* adecuado es fundamental para capturar las características relevantes.

Otro factor que pudo influir en que el *zoom* x10 obtuviera mejores resultados es la variabilidad en el tejido dentro de las anotaciones del patólogo. En ocasiones, el tejido puede no ser uniforme o del mismo tipo, lo que significa que puede haber porciones de tejido que no corresponden a la etiqueta designada. Esto se vuelve más perjudicial con un nivel de *zoom* mayor, ya que la división de los recortes es más fina, lo que puede resultar en recortes que contienen una mayor porción de tejido mal etiquetado. Como consecuencia, el modelo puede enfrentar una mayor confusión al tratar de clasificar estos recortes que contienen información contradictoria. En el caso del *zoom* x10, las divisiones son menos finas y es posible que el modelo encuentre áreas más homogéneas, lo que podría contribuir a obtener mejores resultados[7].

4.2. Modelo en cascada

En este experimento utiliza exclusivamente recortes obtenidos en la resolución de *zoom* x10, ya que en el experimento anterior se obtuvo un mejor resultado con ellos. Las Figuras 4.4 y 4.5 corresponden a las matrices de confusión del modelo de clasificación de tumor/no tumor y del modelo de clasificación de los 5 niveles de reactividad HER2, respectivamente. Están destacados los valores que superan o igualan a los obtenidos por Alegría[7]

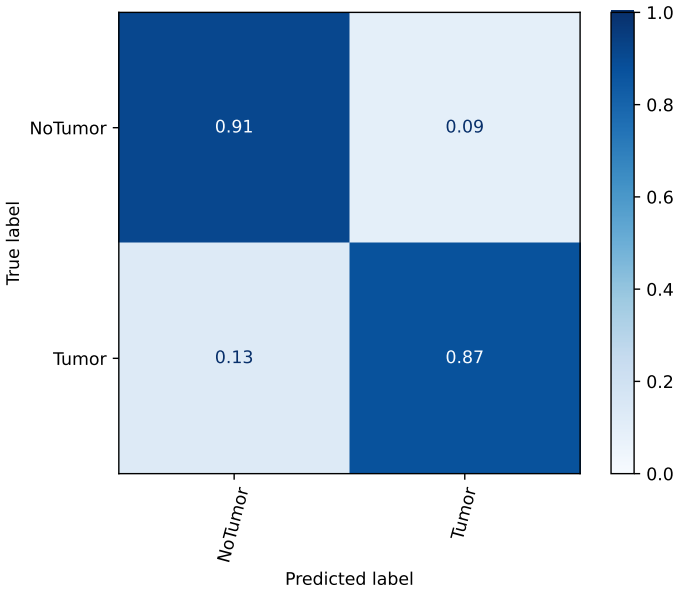


Figura 4.4: Matriz de confusión clasificación de tumor y no tumor con *zoom* x10

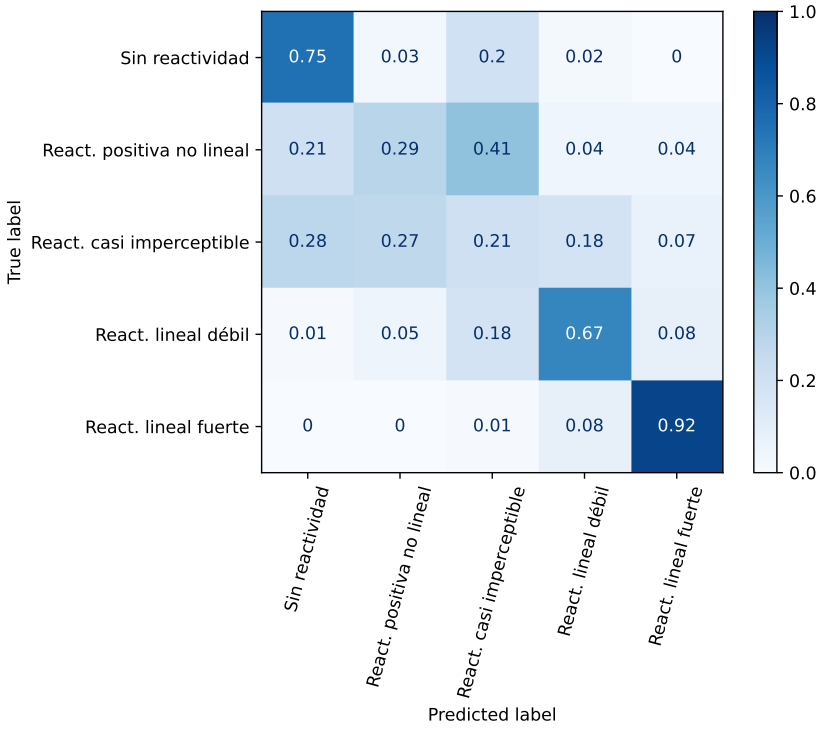


Figura 4.5: Matriz de confusión clasificación de 5 niveles de reactividad con *zoom* x10

Tabla 4.4: Métricas de evaluación para modelo tumor y no tumor

	precisión	recuperación	f1-score	exactitud
NoTumor	0.89	**0.91**	**0.9**	-
Tumor	**0.89**	0.87	**0.88**	-
Promedio macro	**0.89**	**0.89**	**0.89**	**0.89**
Promedio ponderado	**0.89**	**0.89**	**0.89**	**0.89**

Tabla 4.5: Métricas de evaluación para modelo de 5 niveles de reactividad

	precisión	recuperación	f1-score	exactitud
Sin reactividad	**0.85**	**0.75**	**0.79**	-
React. positiva no lineal	**0.39**	0.29	0.33	-
React. casi imperceptible	**0.15**	**0.21**	**0.18**	-
React. lineal débil	0.48	**0.67**	**0.56**	-
React. lineal fuerte	0.79	**0.92**	0.85	-
Promedio macro	0.53	0.57	0.54	**0.61**
Promedio ponderado	0.64	**0.61**	0.62	**0.61**

Si se comparan las métricas de los modelos *Transformer* para la clasificación en cascada de las Tablas 4.4 y 4.5 con sus respectivas contra partes de redes convolucionales de Alegría[7] de las Tablas 2.4 y 2.5, se observa un leve mejor desempeño en los modelos *Transformer*.

Dado que la diferencia de desempeño entre los clasificadores de tumor/no tumor es pequeña, para una implementación real, se podría considerar más conveniente utilizar la red convolucional, ya que, en general, son más livianas y rápidas. Esto se debe a que el modelo de tumor/no tumor tendría que procesar todo el tejido de las biopsias, a diferencia del modelo de tipos de reactividad que solo clasifica las imágenes que fueron categorizadas como tumor. Así, optar por la red convolucional podría ofrecer una mejor eficiencia en tiempo y recursos en el contexto de una aplicación práctica.

Se realizó una evaluación de la clasificación de las 6 clases de ambos modelos trabajando en conjunto. En la Figura 4.6 se muestra su matriz de confusión y en la Tabla 4.6 se muestran sus métricas.

Figura 4.6: Matriz de confusión clasificación 6 clases con modelos cascada funcionando en conjunto.

4.3. Comparación final 6 clases

En este apartado se comparan las métricas de la Tabla 4.6 del clasificador *Transformer* compuesto por 2 modelos en cascada, la Tabla 4.7 del clasificador *Transformer* de 6 clases, y la Tabla 4.8 del clasificador de 6 clases de Alegría[7]. Se han resaltado los mejores valores obtenidos al comparar los 3 casos.

Tabla 4.6: Métricas de evaluación para modelos en cascada unidos para clasificar las 6 clases y con zoom x10

	precisión	recuperación	f1-score	exactitud
No tumor	0.89	**0.91**	0.9	-
Sin reactividad	**0.73**	0.64	**0.68**	-
React. positiva no lineal	**0.37**	0.29	**0.33**	-
React. casi imperceptible	**0.15**	**0.2**	**0.17**	-
React. lineal débil	0.47	**0.56**	**0.51**	-
React. lineal fuerte	0.78	**0.92**	0.84	-
Promedio macro	**0.56**	**0.58**	**0.57**	**0.73**
Promedio ponderado	**0.74**	**0.73**	**0.74**	**0.73**

Tabla 4.7: Métricas de evaluación para modelo de 6 clases y con zoom x10

	precisión	recuperación	f1-score	exactitud
No tumor	**0.91**	0.9	**0.91**	-
Sin reactividad	0.71	0.63	0.66	-
React. positiva no lineal	0.31	0.26	0.29	-
React. casi imperceptible	0.13	0.19	0.15	-
React. lineal débil	**0.48**	0.55	**0.51**	-
React. lineal fuerte	0.8	**0.92**	**0.86**	-
Promedio macro	**0.56**	**0.58**	0.56	**0.73**
Promedio ponderado	**0.74**	**0.73**	0.73	**0.73**

Tabla 4.8: Métricas de evaluación para modelo de 6 clases y con zoom x10 de Alegría[7]

	precisión	recuperación	f1-score	exactitud
No tumor	**0.91**	0.79	0.85	-
Sin reactividad	0.62	**0.71**	0.67	-
React. positiva no lineal	0.25	**0.37**	0.3	-
React. casi imperceptible	**0.15**	0.18	0.16	-
React. lineal débil	0.45	0.37	0.41	-
React. lineal fuerte	**0.81**	0.8	0.8	-
Promedio macro	-	-	-	0.7
Promedio ponderado	**0.74**	0.7	0.72	0.7

Ambos modelos de *Transformer* muestran una mejora respecto el modelo CNN de Alegría[7], pero, según la Tabla 4.6, el clasificador compuesto por dos modelos *Transformer* en cascada obtiene los valores superiores para la mayoría de las métricas.

Es relevante destacar que el modelo *Transformer* en cascada supera a la red convolucional en todas las clases en cuanto al *f1-score*, una métrica que considera tanto la precisión como la recuperación. Además, también lo supera en términos de exactitud. Estos resultados demuestran claramente que el *Transformer* en cascada sobresale por encima de esta red convolucional en especifico en términos de rendimiento.

Es importante mencionar que en el estudio de referencia [7], no se mide el desempeño de los modelos en cascada trabajando en conjunto. Por lo tanto, es posible que ese modelo en cascada hubiese obtenido un rendimiento aún mayor que su clasificador de 6 clases.

Como al comparar las Tablas 4.7 y 4.8 el *Transformer* supera a la red convolucional[7] siendo ambos un modelo de clasificación directa de las 6 clases todo en uno, entonces se puede concluir que el *Transformer* es una herramienta más efectiva para la clasificación de biopsias que la red convolucional.

Capítulo 5

Conclusiones y Trabajo futuro

5.1. Conclusiones

En este trabajo, se ha demostrado que el modelo *Transformer* exhibe un mejor desempeño en general en la clasificación de sobreexpresión de la proteína HER2 en imágenes de biopsias de cáncer gástrico en comparación con la red convolucional. Sin embargo, es importante destacar que este mejor rendimiento no se aplica de manera uniforme a todas las clases. Se puede atribuir este comportamiento a los sesgos inductivos propios de las redes convolucionales, como su invarianza espacial y de traslación, que les otorgan ventaja en ciertos escenarios.

En relación con el experimento en cascada, se ha obtenido una pequeña mejora en el desempeño del modelo. Sin embargo, es posible lograr mayores mejoras mediante el uso de técnicas de *ensemble*, como *Bagging* o *Boosting*. Estas técnicas permiten combinar múltiples clasificadores y mejorar el rendimiento general del modelo, lo que podría resultar en un aumento significativo en la precisión y la capacidad de generalización.

Finalmente, se plantea la posibilidad de que un etiquetado más preciso sobre tejido homogéneo pueda contribuir a obtener mejores resultados con resoluciones más altas. Este enfoque podría ser beneficioso para mejorar la precisión y la capacidad del modelo para distinguir entre las distintas clases de manera más efectiva.

En resumen, el trabajo ha demostrado que el modelo *Transformer* es una herramienta prometedora para la clasificación de sobreexpresión de la proteína HER2 en imágenes de biopsias de cáncer gástrico. Sin embargo, también se reconoce la importancia de explorar y utilizar técnicas adicionales, como *ensemble* y un etiquetado más preciso, para seguir mejorando el desempeño del modelo y abordar los desafíos presentes en esta tarea de clasificación. Estas conclusiones abren el camino para futuras investigaciones y mejoras en el campo de la clasificación de imágenes médicas utilizando redes neuronales del tipo *Transformer*.

El objetivo de esta memoria ha sido alcanzado exitosamente, ya que se logró evaluar el desempeño del modelo *Transformer* como herramienta de clasificación de la reactividad de la proteína HER2, mediante un análisis comparativo con respecto a la red convolucional propuesta por Alegría[7]. Para lograr este resultado, se llevó a cabo un adecuado procesamiento del *dataset*, el cual fue verificado y validado utilizando la aplicación de visualización desarrollada.

5.2. Trabajo futuro

Se sugiere la exploración de diferentes estrategias para mejorar el rendimiento del modelo *Transformer* debido a la gran cantidad de parámetros que posee. Una de las propuestas consiste en desarrollar un pre-entrenamiento personalizado, tomando como base el enfoque planteado en [12], se utilizarían tejidos de biopsias no etiquetados o biopsias sin etiquetar para el proceso de pre-entrenamiento. Esta aproximación permitiría iniciar el entrenamiento de la clasificación con parámetros pre-entrenados, lo que podría acelerar y mejorar el proceso de aprendizaje del modelo.

Adicionalmente, se considera valiosa la incorporación de un término de regularización en la función de costo del modelo, siguiendo la idea presentada en [13]. Esta estrategia de regularización podría contribuir a mejorar la capacidad de generalización del modelo.

Ambas propuestas representan oportunidades prometedoras para optimizar y perfeccionar el desempeño del clasificador *Transformer*, y podrían ser consideradas como temas de investigación futura para mejorar la clasificación de imágenes de biopsias de cáncer gástrico y otros estudios médicos.

Bibliografía

[1] Y.-J. Bang, E. Van Cutsem, A. Feyereislova, H. C. Chung, L. Shen, A. Sawaki, F. Lordick, A. Ohtsu, Y. Omuro, T. Satoh, G. Aprile, E. Kulikov, J. Hill, M. Lehle, J. Rüschoff, Y.-K. Kang, and ToGA Trial Investigators, "Trastuzumab in combination with chemotherapy versus chemotherapy alone for treatment of HER2-positive advanced gastric or gastro-oesophageal junction cancer (ToGA): a phase 3, open-label, randomised controlled trial.," *Lancet (London, England)*, vol. 376, pp. 687–97, 8 2010.

[2] Bettina Müller, "Observational Study of Perioperative Chemotherapy in Gastric Cancer (PRECISO)." https://classic.clinicaltrials.gov/ct2/show/NCT01633203?term=GOCCHI&rank=1. (Consultado 12 ene., 2023).

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," *CoRR*, vol. abs/1706.03762, 2017.

[4] D. Britz, A. Goldie, M.-T. Luong, and Q. Le, "Massive exploration of neural machine translation architectures," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, (Copenhagen, Denmark), pp. 1442–1451, Association for Computational Linguistics, Sept. 2017.

[5] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *ICLR*, 2020.

[6] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, "Transformers in Vision: A Survey," *ACM Computing Surveys*, vol. 54, pp. 1–41, 1 2022.

[7] J. J. Alegría Fuentes, "Clasificación automatizada de sobreexpresión de proteína her2 en biopsias digitalizadas de cáncer gástrico teñidas inmunohistoquímicamente," tesis de magister, Departamento de Ciencias de la Computación, Universidad de Chile, Santiago, 2020.

[8] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, IEEE, 6 2016.

[9] L. F. Escares Garay, "Clasificación de imágenes de cáncer gástrico aplicando aprendizaje profundo," memoria, Departamento de Ingeniería Eléctrica, Universidad de Chile, Santiago, 2020.

[10] T.-T. Wong, "Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation," *Pattern Recognition*, vol. 48, pp. 2839–2846, 9 2015.

[11] R. Wightman, "vit small patch16 224." https://huggingface.co/timm/vit_small_patc

h16_224.augreg_in21k. (Consultado 30 mayo, 2023).

[12] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," *arXiv:2111.06377*, 2021.

[13] A. Steiner, A. Kolesnikov, X. Zhai, R. Wightman, J. Uszkoreit, and L. Beyer, "How to train your ViT? Data, Augmentation, and Regularization in Vision Transformers," *arXiv preprint arXiv:2106.10270*, 6 2021.

Anexos

Anexo A. Gráficos de Exactitud durante el entrenamiento

A.1. Experimento con niveles de *zoom*

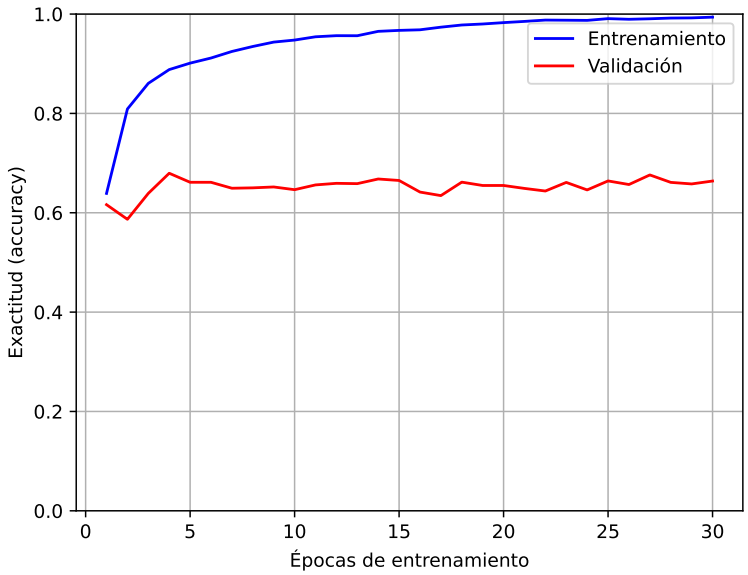


Figura A.1: Gráfico de promedio de exactitud de en clasificación de 6 clases con *zoom* x10

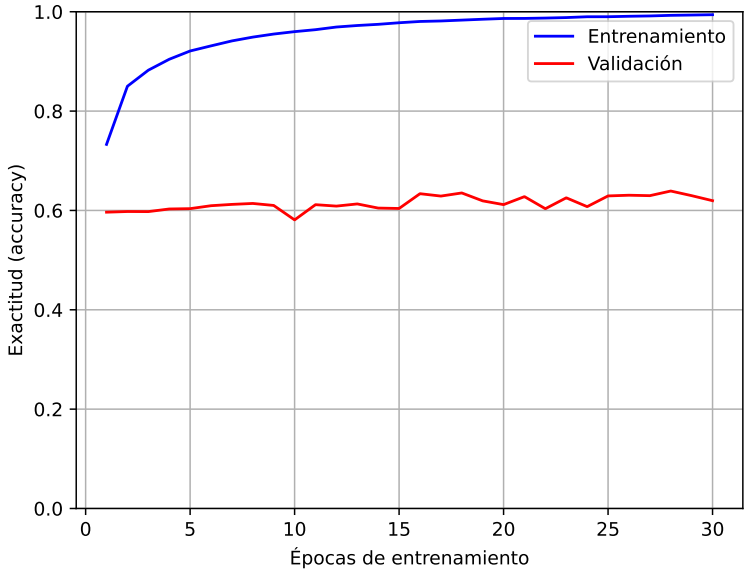


Figura A.2: Gráfico de promedio de exactitud de en clasificación de 6 clases con *zoom* x20

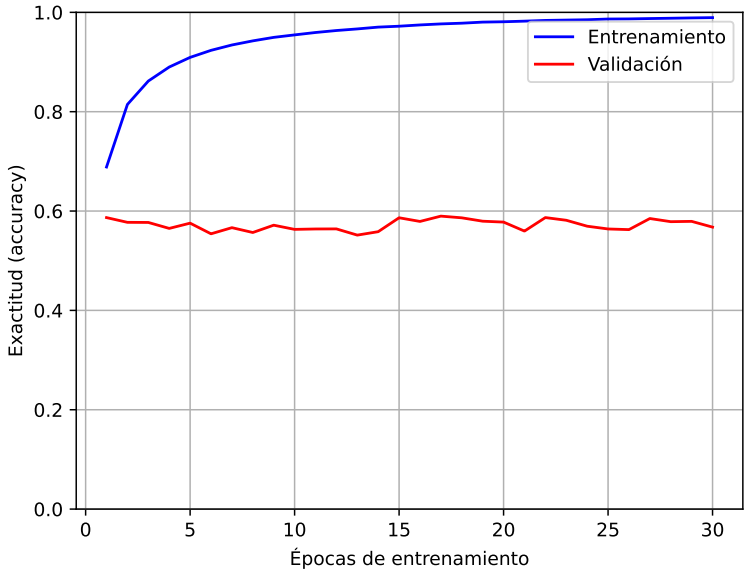


Figura A.3: Gráfico de promedio de exactitud de en clasificación de 6 clases con *zoom* x40

A.2. Modelo en cascada

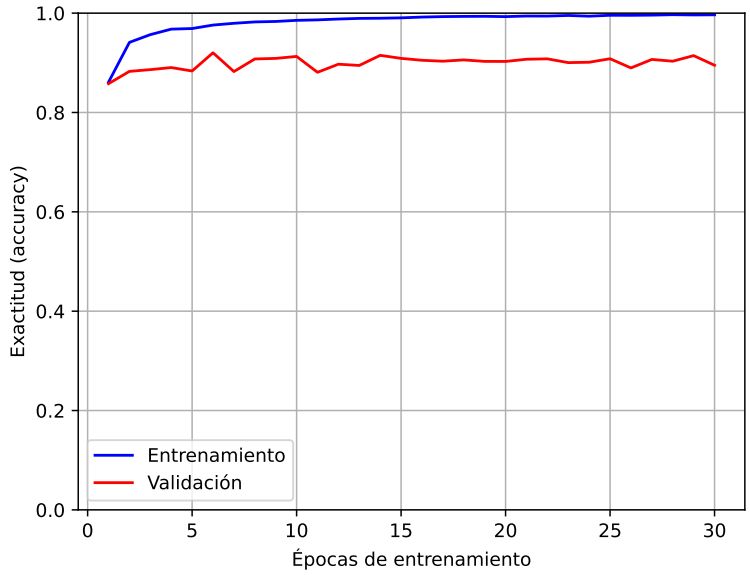


Figura A.4: Gráfico de promedio de exactitud de en clasificación de tumor y no tumor con *zoom* x10

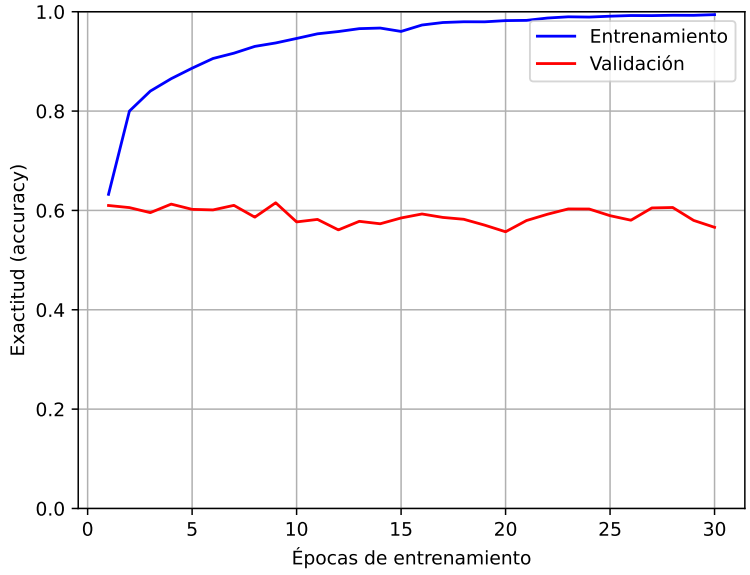


Figura A.5: Gráfico de promedio de exactitud de en clasificación de 5 niveles de reactividad con *zoom* x10