

Comparativa de Redes Neuronales Profundas (RNN, CNN y GNN) para la Clasificación de Tipos de Cáncer usando Datos Multiómicos Integrados.

Figueroa Arcos Arturo^a, Alvarado Martinez Victor M.^b

^a*Centro Nacional de Investigación y Desarrollo Tecnológico, interior internado palmira, Cuernavaca, 62493, Morelos, México*

^b*Centro Nacional de Investigación y Desarrollo Tecnológico, interior internado palmira, Cuernavaca, 62493, Morelos, México*

Abstract

Problema: Destacar la necesidad de métodos precisos para clasificar tipos de cáncer y aprovechar la información de los datos omícos.

Método: Evaluación comparativa de los modelos CNN, RNN y GNN con la integración de datos omícos (mRNA, miRNA y metilación de ADN).

Resultados: Mostrar el rendimiento de los modelos mediante las métricas.

Conclusion: El hallazgo del potencial de las redes neuronales GNN para capturar relaciones biológicas de los datos multiómicos.

Keywords: keyword 1, keyword 2, keyword 3, keyword 4

1. Introducción

El cáncer es una de las principales causas de muerte a nivel mundial, catalogándose como una de las crisis más importantes de salud pública y persistente de la era moderna [1]. Según las estadísticas globales más recientes, en 2020 se registraron casi 10 millones de muertes atribuidas al cáncer, y las estimaciones para el 2022 indican una cifra similar, de cerca 9.7 millones de muertes [2]. A nivel mundial, el cáncer de pulmón, mama y colorrectal son los más prevalentes, que constituyen colectivamente alrededor de dos tercios de los nuevos casos y muertes reportadas en 2022 [3]. Cabe destacar que se ha reportado también un descenso notable en la tasa de mortalidad, con una reducción del 33% debido a los avances en tratamiento y detección temprana,

al igual que varios tipos de cáncer clave están en aumento. Específicamente, se ha observado un incremento anual del 3% en la incidencia de cáncer de próstata entre 2014 y 2019, junto con aumentos continuos en cáncer de mama y de cuerpo uterino [4]. Debido con estos hechos se obtiene una divergencia entre la mejora de la supervivencia y el aumento de la incidencia, con una situación crítica que mientras los avances terapéuticos mejoran la capacidad de tratar la enfermedad establecida, la carga de nuevos diagnósticos no disminuye de manera uniforme. Con este panorama se sugiere que la próxima frontera es el desarrollo de herramientas diagnósticas y de clasificación más precisas y tempranas, que sean capaces de abordar el cáncer en sus etapas moleculares incipientes.

El cáncer es un complejo y heterogéneo conjunto de patologías, debido a un crecimiento celular descontrolado, impulsado por una acumulación de alteraciones genéticas y epigenéticas [5]. La clasificación de tipos de cáncer es fundamental en el contexto de la medicina de precisión, esto permite la selección de terapias dirigidas, estratificación de pacientes y la mejora de resultados de pronóstico [6]. Sin embargo, se tiene un desafío debido a la alta similitud hispatológica entre ciertos tumores y la heterogeneidad intratumoral.

Históricamente, la investigación del cáncer se ha centrado en capas individuales de datos biológicos, con mayor frecuencia los datos de expresión génica (transcriptómicos). Sin embargo, una sola capa no es capaz de capturar la complejidad del cáncer, presentando una versión incompleta y en ocasiones, engañosa de la biología tumoral. Este enfoque a menudo solo suele revelar un tipo de información omica a la vez [7]. De la misma manera, el análisis de una sola omica suele presentar sesgos inherentes a la tecnología y los hallazgos al estudiar una sola capa reduce la de la validación [8].

Para contrarrestar esta limitante y capturar una visión holística de la biología del cáncer, se ha precisado la integración de múltiples conjuntos de datos ómicos generados a partir de los mismos pacientes. El enfoque de integración multiómica permite descubrir patrones biológicos más complejos y obtener una comprensión funcional completa [9]. La combinación de datos omicos permite la identificación de subtipos moleculares distintos que podrían permanecer sin ser detectados con el análisis de una sola omica, proporcionando una caracterización más profunda y precisa de la enfermedad de un paciente [10].

La integración de datos multiómicos a pesar de su potencial, presenta desafíos computacionales. Estos datos presentan el problema de la alta dimensionalidad (el " problema de $p \gg n$ ", donde el número de características p , supera con creces el número de muestras n) lo que aumenta el riesgo de sobreajuste (overfitting). Además, los datos ómicos son inherentemente ruidosos y a menudo, incompletos. Sin un preprocessamiento riguroso, los modelos de aprendizaje profundo pueden fallar debido al vasto espacio de características y ruido debido a su heterogeneidad y la presencia de complejas interacciones no lineales

entre las diferentes capas biológicas [11]. Si bien los métodos clásicos de aprendizaje automático (como SVM o Random Forest) han demostrado cierto éxito, a menudo luchan por modelar eficazmente estas no linealidades y la jerarquía de las interacciones biológicas.

En este contexto, el Aprendizaje Profundo (DL) surge como una poderosa herramienta del subcampo de las redes Neuronales Artificiales (ANN), capaz de aprender de representaciones jerárquicas y patrones complejos a partir de datos de alta dimensionalidad [12]. La literatura reciente ha demostrado la aplicación exitosa de modelos de DL en una variedad de tareas oncológicas, como la clasificación de tipos y subtipos de cáncer, identificación de genes conductores y predicción de la supervivencia utilizando datos omicos de fuentes como TCGA [13].

El campo del DL no está definido por un solo modelo, si no por una comparación de arquitecturas para identificar cuál es la que se adapta mejor a la estructura de datos biológicos [13]. Se han aplicado diversas arquitecturas a los datos omicos como por ejemplo las Redes Neuronales Convolucionales (CNN), diseñadas originalmente para el procesamiento de imágenes, se han adaptado para encontrar patrones locales en los datos genómicos, a menudo tratando el vector de características ómicas como una imagen de 1D o una matriz de características de 2D [14, 15]. Por su parte, las Redes Neuronales Recurrentes (RNN), como las LSTM o GRU, están diseñadas para datos secuenciales y pueden, en teoría, capturar dependencias a largo plazo dentro del vector de características concatenadas, aunque esta representación secuencial es una abstracción de la biología subyacente [16, 17].

Sin embargo, en las CNN como en las RNN se tienen la limitación que se tratan las características ómicas (genes, miARNs, sitios CpG) como una secuencia o cuadrícula arbitraria. Es decir, que ignoran en gran medida la topología biológica inherente, que los genes y sus productos no actúan de forma aislada, sino dentro de redes de interacción complejas (rutas metabólicas, redes de regulación génica, interacciones proteína-proteína).

Las Redes Neuronales de Grafos (GNN) son una opción que ofrece una ventaja conceptual decisiva. Las GNN son una arquitectura que aprendizaje profundo que están diseñadas para procesar datos que están estructurados como grafos [18, 19]. A diferencia de las CNN o RNN, el sesgo inductivo de una GNN es la conectividad relacional. Su poder reside en su capacidad para modelar dependencias complejas y capturar relaciones de orden superior de las redes biológicas mediante la agregación iterativa de información de los nodos vecindarios de los nodos. Al tener las características ómicas y representarlas como nodos en un grafo y sus interacciones conocidas (o inferidas) como ejes, las GNN son capaces de aprender de las representaciones de los nodos agregando información de sus vecindarios biológicamente relevantes.

El objetivo de este trabajo es hacer una comparación del rendimiento de las arquitecturas CNN, RNN y GNN en una tarea de clasificación de 31 tipos de cáncer y muestras normales, utilizando un único conjunto de datos multiómicos integrados, que fueron preprocesados con un análisis de DGE/CpG para selección de genes relevantes, y selección y reducción de características mediante LASSO, la estrategia de representación de los datos fue de matriz de 2D para los modelos CNN, vector de 1D para el modelo RNN y grafo de correlación para los modelos GNN.

2. Metodología

2.1. Recopilación de datos

Los datos multiómicos de los 31 tipos de cáncer utilizados en este trabajo se tomaron de proyecto Pan-cáncer del Atlas del Genoma de Cáncer (TCGA) [20]. Utilizando la biblioteca TCGAbiolinks en R que realiza consultas con la herramienta GDC (Genomic Data Commons), el cual es parte de un proyecto dedicado a proporcionar una base de datos centralizada para proyectos de investigación en estudios genómicos del cáncer [21]. Los tipos de cáncer de los tumores y las muestras normales por cada muestra de miARN, ARNm y metilación de ADN se muestran en la siguiente Figura 1. El conjunto inicial total consta de 10,511 muestras de ARNm, 10,048 muestras de miARN y 8,696 muestras de metilación de ADN.

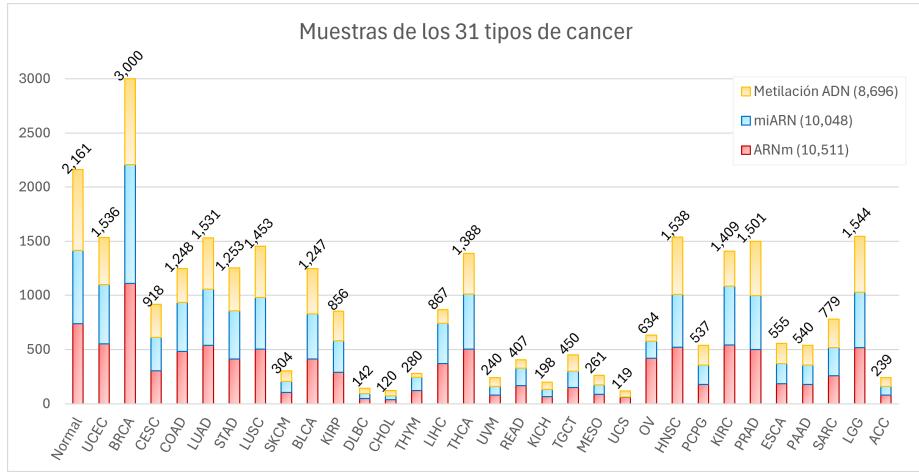


Figure 1: Número datos multiómicos total (ARNm, miARN y metilación de ADN) y su distribución por cada tipo muestra tumoral y normal, incluidos en el proyecto Pan-cáncer de los 31 tipos de cancer

Los datos en bruto obtenidos cuentan con características (genes) para cada dato ómico, para los datos de ARNm se tienen 60,660 características, para los datos de miARN se tienen 1,881 características y para los datos de metilación de ADN se tienen 485,577 características como se muestra en la Figura 2.

60,660 genes							1,881 genes						
ARNm	Gene 1	Gene 2	Gene 3	Gene 4	Gene ...	Gene 60,660	miARN	Gene 1	Gene 2	Gene 3	Gene 4	Gene ...	Gene 1,881
BRCA	2080	2993	2449	1582	846	100	BRCA	2319	744	1417	1177	860	489
Normal	2213	603	1573	2460	120	2496	Normal	1642	2750	1392	676	2070	849
OV	280	1001	217	1892	2588	1048	OV	285	2373	607	880	390	705
LUNG	2406	2451	2333	182	2633	1975	LUNG	2908	1328	2488	1252	1883	2220
CESC	145	1514	1554	1904	246	2324	CESC	1333	239	883	321	1608	2581
BLCA	1840	2383	1102	2011	457	2633	BLCA	1943	171	52	837	1390	2410
KICH	1854	874	91	382	1967	1845	KICH	1912	2184	536	1756	440	563
LIHC	1564	1204	2726	399	1883	172	LIHC	2833	2386	456	1781	757	492

485,577 genes						
Metilacion ADN	Gene 1	Gene 2	Gene 3	Gene 4	Gene ...	Gene 485,577
BRCA	859	493	1357	2825	2212	2142
Normal	968	2856	813	1288	898	839
OV	2337	222	2913	769	1169	894
LUNG	2276	2076	1248	2133	595	2903
CESC	1960	1298	561	2032	2170	1792
BLCA	2536	2568	145	928	201	479
KICH	1632	1784	374	2382	1004	1946
LIHC	969	714	1588	2015	2953	865

Figure 2: Número datos multiómicos total (ARNm, miARN y metilación de ADN) y su distribución por cada tipo muestra tumoral y normal, incluidos en el proyecto Pan-cáncer

2.2. Preprocesamiento de datos

Los datos en bruto inicial, abarca decenas de miles de genes y sitios CpG, que contienen una gran cantidad de variables no informáticas o ruidosas. Para mitigar la alta dimensionalidad y mejorar la convergencia de los modelos de DL, se aplicó una estrategia de filtrado jerárquico.

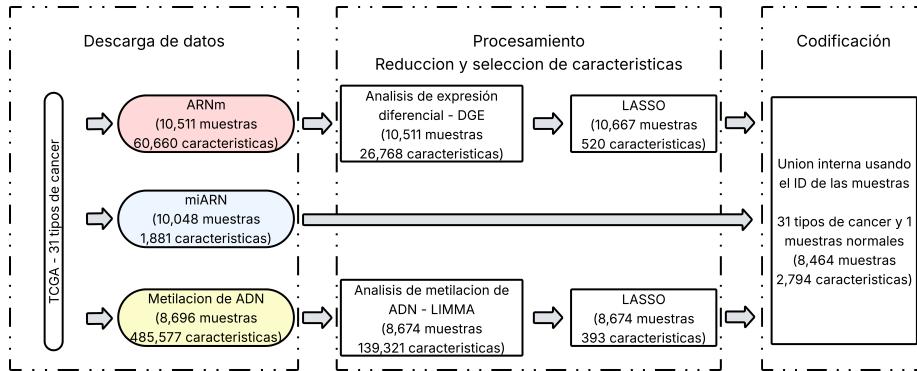


Figure 3: Número datos multiómicos total (ARNm, miARN y metilación de ADN) y su distribución por cada tipo muestra tumoral y normal, incluidos en el proyecto Pan-cáncer

El preprocesamiento de datos en bruto se realizó la identificación de características biológicas relevantes y la reducción de dimensionalidad, para extraer características biológicas importantes de datos de alta dimensionalidad [22]. Para la identificación biomarcadores que muestran diferencias significativas en los datos de ARNm y metilación se utilizó el análisis de expresión diferencial (DGE) y análisis de metilación diferencial con la herramienta LIMMA. El análisis DGE se utilizó con el paquete de R DESeq2, que consiste en modelar los

datos de ARNm usando distribución binomial negativa, para identificar genes que presentan cambios significativos en los niveles de expresión génica [23]. LIMMA se utilizó con el paquete en R, que asume que los datos siguen una distribución normal (Gaussiana) para identificar sitios CpG metilados diferencialmente significativos [24]. Posteriormente, LASSO fue aplicado en las características resultantes del análisis diferencial para extender la selección de características de los datos de ARNm y metilación de ADN y reducir las [25], en la Figura 3 se muestra en flujo del procesamiento de los datos.

2.2.1. Análisis de expresión génica diferencial (DGE)

El Análisis de Expresión Diferencial (DGE) es una metodología fundamental en genómica para comparar los niveles de expresión génica bajo diferentes condiciones biológicas (ej. tratamiento vs. control, o tejido normal vs. tejido tumoral). Esta herramienta permite identificar y caracterizar los genes que modifican su actividad entre la condición de referencia y la experimental.

Dado que los datos de secuenciación (ARNm) presentan una naturaleza de conteo discreto y exhiben sobredispersión, se utilizó el paquete DESeq2. Este método proporciona un marco estadístico robusto basado en Modelos Lineales Generalizados (GLM) que asumen una distribución binomial negativa para los conteos de lecturas. La evaluación de la significancia estadística de los cambios en la expresión entre muestras de tejido normal y muestras de tejido tumoral se realizó mediante la prueba de Wald. Para el criterio de selección, se definieron como diferencialmente expresados únicamente aquellos genes que cumplieron con una significancia estadística estricta, definida por un umbral de valor p ajustado (FDR) ≤ 0.001 .

2.2.2. Análisis de metilación diferencial

Para la identificación de alteraciones epigenéticas significativas, se utilizó la herramienta LIMMA (Linear Models for Microarray Data), mediante la cual se ajustó un modelo lineal a los niveles de metilación de cada sitio CpG en función de los grupos experimentales (muestras tumorales vs. normales). El conjunto de datos analizado comprendió 9,171 muestras y 485,577 características obtenidas de la plataforma Human Methylation 450K (HM450). Este enfoque estadístico emplea el método Bayesiano empírico para calcular estadísticas *t* moderadas, lo que permite evaluar la asociación entre el estado de metilación y el fenotipo de manera robusta al estabilizar los errores estándar a través de los miles de sitios analizados.

La selección de los sitios CpG diferencialmente metilados se realizó aplicando un criterio de significancia estadística con un valor p de corte de 0.05. Este procedimiento permitió filtrar los datos originales y reducir el espacio de características a 139,321 sitios CpG, concentrando el análisis en los cambios de metilación más significativos. De estos candidatos, se conservan para la etapa de integración multiómica aquellos que mostraron una diferencia relativa sustancial en los valores Beta, asegurando que las características seleccionadas

representen cambios epigenéticos con un potencial impacto funcional en los mecanismos de tumorigénesis.

2.2.3. LASSO

Después del filtrado inicial mediante análisis diferencial, se procedió a una reducción de dimensionalidad utilizando la regresión LASSO (Least Absolute Shrinkage and Selection Operator). El método se aplicó de manera independiente a las matrices de datos ARNm y metilación de ADN, permitiendo una selección de características optimizada para la escala y distribución específica de cada perfil ómico antes de su integración. Para cada conjunto de datos, se ajustó el hiperparámetro de regularización (λ) mediante validación cruzada, seleccionando el valor óptimo que minimizó el error de predicción para identificar el subconjunto más robusto de biomarcadores.

Matemáticamente, el algoritmo selecciona las características minimizando la suma de los residuos cuadrados sujeta a una penalización en la norma ℓ_1 de los coeficientes, definida por la siguiente función objetivo como:

$$\min_{\beta} \left\{ \sum_{i=1}^M \left(y_i - \sum_{j=1}^p x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^p |\beta_j| \right\} \quad (1)$$

donde M representa el número de muestras, p el número de características de entrada, y λ controla la intensidad de la penalización. Al forzar que los coeficientes de las variables menos relevantes converjan a cero, la aplicación de LASSO actúa como un filtro, únicamente las características con coeficientes $\beta_j \neq 0$ se conservaron.

2.2.4. Integración y construcción de matriz multiómica

Tras la selección de características independiente, se procedió a la fusión de los datos de ARNm, miARN y metilación en una estructura de datos unificada. La integración se realizó mediante una operación de unión interna (inner join) basada en los identificadores únicos del paciente, asegurando la correspondencia exacta de las tres modalidades ómicas para cada individuo y excluyendo aquellas muestras que no presentaran información completa.

El conjunto de datos final consolidado (matriz multiómica) quedó constituido por un total de 8,464 muestras y 2,794 características biológicas tras el filtrado. Este registro unifica la diversidad fenotípica de las 32 clases objetivo (31 tipos de cáncer y tejido normal), el cual servirá con entrada estandarizada para el entrenamiento y evaluación comparativa de las arquitecturas de aprendizaje profundo.

2.3. Estructuración de datos para modelos de aprendizaje profundo

A partir de la matriz multiómica, constituida por 8,464 muestras y un vector de características de dimensión $N = 2,794$, se generaron tres representaciones de datos distintas. Esta etapa fue fundamental para adaptar la entrada a los requerimientos estructurales de cada arquitectura de aprendizaje profundo evaluada (espacial, secuencial y relacional).

2.3.1. Estructuración espacial 2D (CNN)

Para la implementación de las Redes Neuronales Convolucionales (CNN), el vector de características unidimensional de cada muestra se transformó en una estructura matricial bidimensional, simulando el formato de una imagen. Dado que la dimensión original no corresponde a un cuadrado perfecto, se calculó la dimensión entera más próxima capaz de contener la totalidad de los datos (matriz de 53×53). Las posiciones vacías restantes en la cuadrícula se completaron mediante técnica de relleno con ceros (zero-padding). En esta disposición, cada pixel representa el nivel de expresión o metilación de una característica específica; sin embargo, es importante notar que la vecindad espacial resultante a causa del reordenamiento no refleja necesariamente proximidad biológica.

2.3.2. Estructuración secuencial (RNN)

En la evaluación de los modelos recurrentes mediante la integración de datos multiómicos, se utilizaron dos estrategias para la secuenciación de los datos:

A. Secuenciación de características globales

Los datos se tomaron de la matriz y reformateados en un vector concatenado de $N = 2,794$ características como una secuencia temporal $S_{global} = \{f_1, f_2, \dots, f_N\}$. El vector se introdujo al modelo paso a paso. Esta representación asume una dependencia ordinal entre las características individuales. Este enfoque evalúa la capacidad bruta de la RNN para memorizar dependencias de largo alcance en series ruidosas.

B. Estructuración Jerárquica por Modalidad

Los datos se organizaron en bloques lógicos, con el fin de facilitar la captura de las interacciones biológicas a diferencia de usar una lista plana y larga de números. El proceso fue el siguiente:

Compresión por tipo de ómica: Los datos se separaron en sus grupos originales: ARNm (x_{mrna}), miARN (x_{mirna}) y metilación (x_{meth}). Cada grupo se procesó de forma independiente a través de capas densas para extraer sus características más relevantes, esta información se guardó en un resumen compacto (embedding) de 512 valores (dimensión $d = 512$), generando tres vectores ($e_{mrna}, e_{mirna}, e_{meth}$).

Creación de secuencia: Los vectores resultantes se organizaron para formar una secuencia de solo 3 pasos de tiempo ($T = 3$). El orden se definió siguiendo el flujo de la biología molecular, empezando por los datos de metilación, seguidos del miARN y por último el ARNm [26, 27]. Los embeddings se apilaron para formar la secuencia temporal $S_{latent} = [e_{meth}, e_{mirna}, e_{mrna}]$.

La secuencia resultante se utilizó como entrada al modelo RNN. Con lo que se espera que aprenda la influencia de cada nivel ómico, en lugar de intentar encontrar patrones en una secuencia larga de variables individuales.

2.3.3. Estructuración basada en grafos (GNN)

La diferencia con los enfoques anteriores, para las GNN se construyó una estructura topológica basada en relaciones biológicas inferidas. Se definió un grafo $G = (V, E)$, donde el conjunto de los nodos V corresponde de las 2,794 características ómicas. Para definir la conectividad o aristas (E), se evaluó la dependencia lineal por pares entre todas las características utilizando el conjunto multiómico. La métrica utilizada fue el coeficiente de correlación de Pearson (r), el cual se calculó para dos vectores de características S_i y S_j de acuerdo con la ecuación siguiente:

$$\text{corr}(S_i, S_j) = \frac{\text{cov}(S_i, S_j)}{\sigma_{S_i} \sigma_{S_j}} \quad (2)$$

Donde cov representa la covarianza y σ la desviación estándar de las características respectivas. La matriz de adyacencia A resultante se construyó binarizando estas relaciones; se estableció una conexión física entre nodos i y j únicamente si presentaban una correlación fuerte, aplicando un umbral de corte de $r \leq -0.8$ o $r \geq 0.8$. De esta manera, la arquitectura GNN recibe dos entradas: La matriz global de características (atributos del los nodos) y la matriz de conexiones o aristas (matriz de adyacencia). Esta configuración permite que la arquitectura realice operaciones de paso de mensajes (message passing), agregando información de características funcionalmente correlacionadas independientemente de su posición secuencial en el vector de datos original.

2.4. Arquitecturas de los modelos de aprendizaje profundo

Se diseñaron e implementaron tres arquitecturas de redes neuronales distintas para abordar la tarea de clasificación multiclas. Aunque las estrategias de extracción de características variaron, todas las redes compartieron un bloque clasificador final.

2.4.1. Modelos convolucionales

Para evaluar el rendimiento del aprendizaje de características espaciales sobre lo mapas multiómicos artificiales (53×53), se implementaron dos modelos distintos, el primero una variante del modelo convolucional LeNet-5 y el segundo

una arquitectura profunda con normalización por lotes. Ambas modelos buscan identificar patrones locales en la matriz reordenada, aunque con diferentes profundidades y funciones de activación.

A. Variante de LeNet-5

Este modelo se adaptó de la clásica LeNet-5, que su propósito original para el que se diseñó es para el reconocimiento de dígitos [28], para procesar la complejidad de los datos omicos. El modelo consta de dos bloques convolucionales secuenciales que funcionan como capas de extracción, el primer bloque con 6 filtros (5×5) y el segundo con 16 filtros (5×5) la configuración se muestra en la Figura 4. La función de activación utilizada fue la tangente hiperbólica (tanh) y reducción de dimensionalidad Average Pooling (2×2), preservando el flujo de información promedio de los vecindarios de características. Seguido de estos bloques se aplano la salida y se sometió a un clasificador denso, dada la alta dimensionalidad de los datos de entrada, se utilizaron 4 capas ocultas consecutivas de 1000, 500, 120 y 84 neuronas, todas con activación tanh y por último una capa de clasificación con activación Softmax. Entre cada capa de convolución y densa se aplicó Dropout con una tasa de 0.1 para mitigar el sobreajuste, el optimizador usado fue Adam y un tamaño de lote (batch size) de 128.

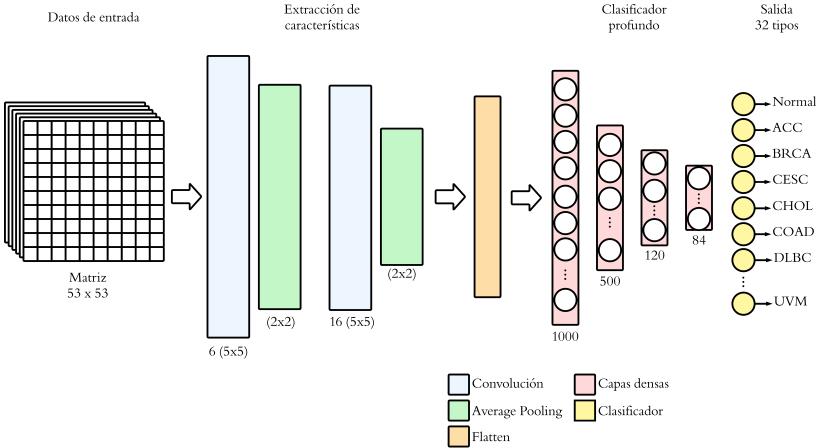


Figure 4: Adaptación de la arquitectura LeNet-5. Esquema de la red convolucional jerárquica que ilustra la transición desde la matriz de entrada (53×53) hacia la extracción de características mediante dos bloques de filtros ($6 \times 5 \times 5$ y $16 \times 5 \times 5$). El diagrama detalla la integración de Average Pooling (2×2) y la fase de aplanamiento (Flatten) previa a un clasificador denso de cuatro capas ($1000 \rightarrow 500 \rightarrow 120 \rightarrow 84$) con activaciones tanh, culminando en la capa Softmax para la clasificación de los 32 subtipos tumorales.

CNN profunda con normalización

El segundo modelo se adaptó de acuerdo con el propuesto por Chuang et al.

(2021) [15], la cual demostró una alta eficacia en la predicción de tipos de cáncer mediante la integración de datos ómicos y redes de interacción. Este modelo consta de tres bloques de extracción de características diseñados para capturar patrones no lineales complejos (Figura 5). Cada uno de los bloques está compuesto secuencialmente por una capa de convolución de 64 filtros, normalización por lotes (Batch Normalization) para estabilizar la distribución de las activaciones, una capa de Max-Pooling (2×2) y Dropout. El primer bloque utiliza un campo receptivo mayor con kernel de 5×5 , mientras que los dos bloques subsiguientes refinan las características con kernels de 3×3 con activación ReLU para mitigar el problema del desvanecimiento del gradiente. Posterior a las capas de convolución, se tiene una capa de aplanamiento seguido de capas profundas de 1000, 600 y 80 neuronas antes de la capa de clasificación final con función Softmax. El optimizador utilizado fue Adam con una tasa de aprendizaje de 1×10^{-4} y un tamaño de lote reducido de 24.

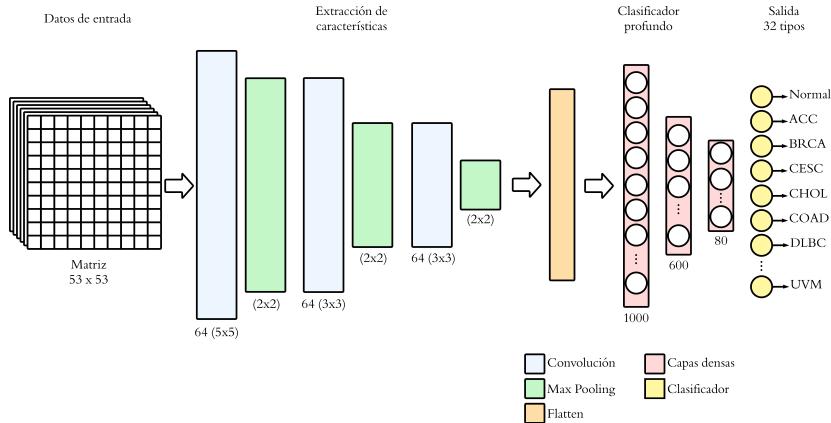


Figure 5: Arquitectura de modelo CNN profunda. Esquema basado en la propuesta de Chuang et al. para la captura de patrones no lineales. El diagrama ilustra tres bloques de extracción de características con filtros de 64 y reducción mediante Max-Pooling (2×2), transitando desde kernels de 5×5 hacia 3×3 . Se detalla el proceso de aplanamiento (Flatten) y la transición hacia un clasificador profundo de tres capas densas ($1000 \rightarrow 600 \rightarrow 80$) para la distinción de los 32 subtipos tumorales mediante una capa final Softmax.

2.4.2. modelos RNN

Para evaluar la capacidad de los modelos recurrentes en la integración de datos multiómicos, se contrastaron dos estrategias de modelos secuenciales, un modelo de referencia basado en LSTM apiladas (para secuencias de características planas) y una arquitectura jerárquica multimodal (basada en GRU bidireccional).

Modelo LSTM Apilado

Como primera estrategia, se implementó una modelo con arquitectura profunda basada en unidades de memoria de corto y largo plazo (LSTM) como se

muestra en la Figura 6. Este modelo aborda el vector de características temporales completo como una única secuencia temporal continua. La extracción de características temporales se realiza mediante dos capas LSTM consecutivas: una capa inicial de 128 unidades configurada para retornar la secuencia completa, seguida de una segunda capa de 64 unidades que condensa la información en un vector de contexto global. La etapa de clasificación está compuesta por una red totalmente conectada de cuatro capas densas decrecientes de 1000, 500, 200 y 60 neuronas, la cual fue diseñada para modelar interacciones altamente no lineales a partir del estado oculto de la RNN, en ambas etapas se utilizó la función de activación tanh. Por último se tiene una capa de clasificación con función de activación Softmax.

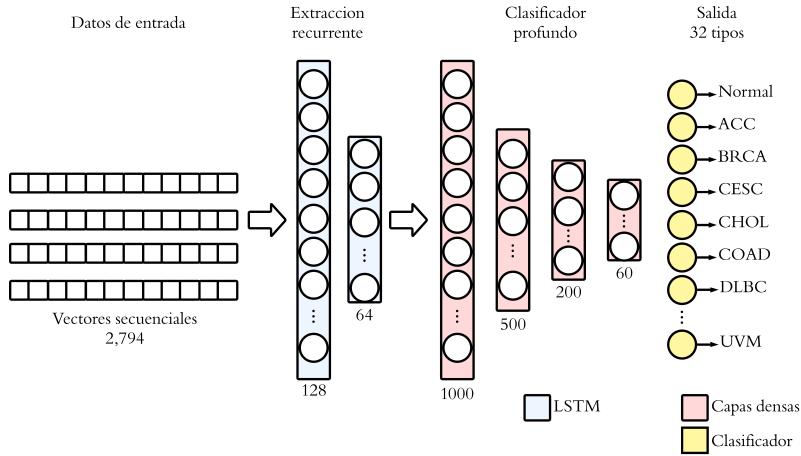


Figure 6: Arquitectura de modelo LSTM apilado. Esquema del modelo de procesamiento secuencial para el vector de 2,794 características. El diagrama ilustra la etapa de extracción recurrente mediante dos capas LSTM (128 y 64 unidades) para la generación de un vector de contexto global. Se detalla el clasificador profundo compuesto por cuatro capas densas ($1000 \rightarrow 500 \rightarrow 200 \rightarrow 60$) con activación tanh, culminando en la capa de salida Softmax para la clasificación de los 32 tipos tumorales.

Modelo RNN Multimodal Jerárquico

Por otro lado, para mitigar los problemas de dispersión de señal en secuencias largas, se implementó una variante jerárquica optimizada denominada OmicsRNN (Figura 7). A diferencia del modelo anterior, este modelo separa inicialmente los datos en sus tres modalidades constitutivas (ARNm, miARN y metilación), en vectores latentes de 512 dimensiones mediante proyecciones densas independientes (Lineal \rightarrow Batch Normalization \rightarrow ReLU \rightarrow Dropout). Esto permite al modelo aprender representaciones específicas para cada ómica antes de la integración.

Posteriormente, estos embeddings latentes se organizaron en una secuencia corta de tres pasos ($T = 3$) siguiendo el flujo de regulación biológica (Meti-

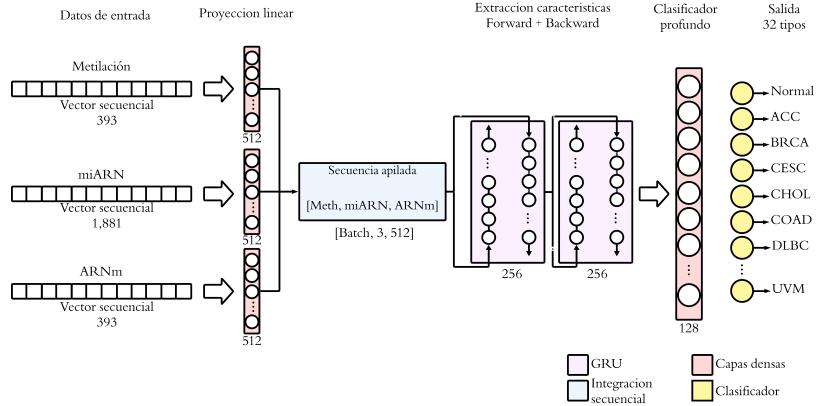


Figure 7: Arquitectura de modelo RNN jerárquico. Esquema del modelo basado en el flujo de regulación biológica. Se ilustra la proyección lineal independiente de las tres modalidades (393 Metilación, 1,881 miARN, 393 ARNm) hacia subespacios latentes de 512 dimensiones. El diagrama detalla la construcción de la secuencia apilada ($T = 3$) y su procesamiento mediante una red recurrente bidireccional de dos capas (256 unidades), finalizando en un clasificador profundo de 128 neuronas para la distinción de los 32 subtipos tumorales.

lación → miARN → ARNm). La secuencia compacta alimenta una Unidad Recurrente Con Puerta (GRU) bidireccional de 2 capas con 256 unidades ocultas. La bidireccionalidad es crucial, ya que permite al modelo capturar tanto la regulación hacia adelante como las dependencias inversas o de retroalimentación biológica. Finalmente, el estado oculto concatenado de la GRU se procesa mediante un clasificador de una única capa densa de 128 neuronas con activación ReLU y regularización Dropout.

2.5. Modelos GNN

Para explotar la topología relacional definida por la matriz de correlación de Pearson, se implementó un marco de modelado basado en grafos utilizando la biblioteca Pytorch Geometric. El grafo de entrada, definido por la matriz de características X y la lista de aristas (edge index), fue procesado por tres arquitecturas distintas: una convolucional grafica, una basada en atención jerárquica y una basada en transformadores, esto para evaluar el impacto de diferentes mecanismos de agregación de vecindad (message passing) en la tarea de clasificación multiclas.

C. GCN (*Graph Convolutional Network*)

Se diseñó una GCN de 4 capas con una arquitectura de compresión piramidal (Figura 8). A diferencia de los modelos de ancho constante, esta variante fuerza al modelo a sintetizar las características más relevantes reduciendo la dimensionalidad del espacio latente en cada salto del grafo. La primera capa convolucional proyecta el vector de entrada a un espacio oculto de 600 dimen-

siones. Las capas subsiguientes aplican una reducción progresiva a 300 y 150 dimensiones respectivamente, obligando a la red a aprender representaciones cada vez más abstractas y compactas de los vecindarios biológicos. Finalmente, la cuarta capa proyecta el vector de 150 características hacia las 32 clases de salida. Cada etapa intermedia utiliza la función de activación ReLU y una regularización por Dropout del 0.3 para mitigar el sobreajuste durante la compresión de características.

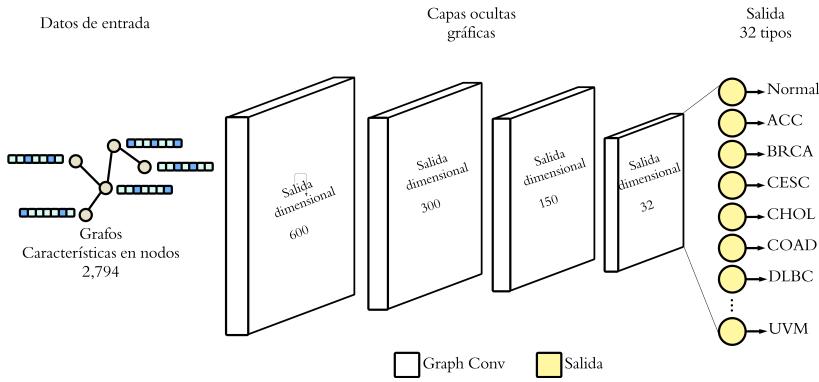


Figure 8: Arquitectura GCN (Graph Convolutional Network). Flujo de procesamiento desde la entrada multi-ómica ($N = 2,794$) hasta la clasificación de 32 tipos de cáncer. Los bloques representan la reducción jerárquica de dimensionalidad ($600 \rightarrow 300 \rightarrow 150 \rightarrow 32$), diseñada para la síntesis de representaciones latentes abstractas. Se indican las capas de convolución gráfica (bloques blancos) y la capa de salida (esferas amarillas), integrando regularización Dropout y activations ReLU en cada etapa de compresión.

A. GAT (Graph Attention Network)

La GAT se implementó con cuatro capas diseñadas con un esquema de atención piramidal, diseñada para gestionar la alta complejidad de las interacciones genéticas (Figura 9). La primera capa proyecta las características de entrada a un espacio latente de 1024 dimensiones distribuidas en 8 cabezales de atención, generando un intermedio masivo de 8,192 características por nodo. Para canalizar esta información, las capas subsiguientes reducen progresivamente la dimensionalidad (512 características \times 4 cabezales) y la tercera a 512 dimensiones (256 características \times 2 cabezales). Finalmente, la última capa consolida la información en un único cabezal que mapea directamente a las 32 dimensiones de salida correspondiente a las clases objetivo. Este flujo se estabiliza mediante normalización por lotes aplicada a las salidas de alta dimensionalidad antes de la activación LeakyReLU y Dropout del 0.1.

B. GTN (Graph Transformer Network)

Para capturar dependencias globales complejas, se evaluó una arquitectura basada en Graph Transformer que mantiene una densidad de características

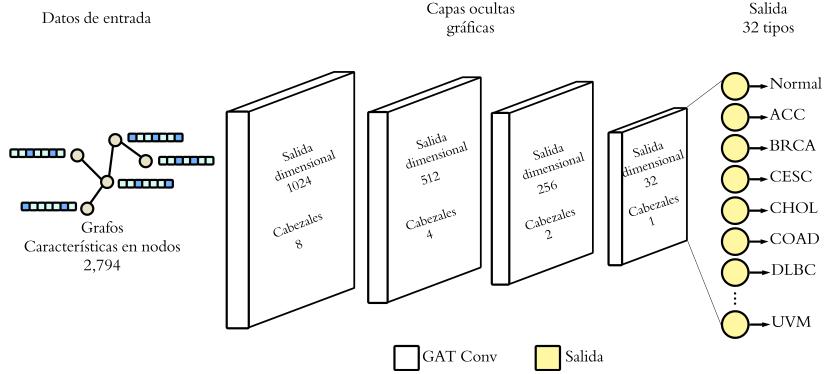


Figure 9: Arquitectura GAT (Graph Attention Network). Representación del flujo de atención multi-cabezal para la clasificación de 32 tipos de cáncer. El diagrama detalla la arquitectura de compresión jerárquica mediante capas GATConv, partiendo de una proyección inicial de alta densidad (1024×8 cabezas) hacia capas de dimensiones reducidas (512×4 y 256×2). Las esferas amarillas representan la consolidación final en un único cabezal de atención para el mapeo de las clases objetivo. Se indica la integración de Batch Normalization y activación LeakyReLU para la estabilidad del entrenamiento.

constante a través de sus capas profundas (Figura 10). El modelo consta de tres bloques TransformerConv consecutivos, configuradas cada uno con 6 cabezas de atención y 80 canales por cabezal, resultando en una representación vectorial constante de 480 dimensiones por nodo a lo largo de la red. Esta preservación del ancho de banda permite al modelo refinar las representaciones sin pérdida de información prematura. La capa final transforma este vector de 480 dimensiones mediante un único cabezal de atención hacia los 32 logits de clasificación. Dado el alto número de parámetros implicados en los bloques de atención densa, se aplicó una tasa de Dropout del 0.5 junto con activations LeakyReLU para maximizar la capacidad de generalización.

2.6. Configuración experimental y métricas de evaluación

La evaluación comparativa de las arquitecturas propuestas se fundamentó en un marco metodológico estricto, diseñado para maximizar la transparencia y la solidez estadística del estudio. Esta sección describe el entorno de hardware y software implementado, así como las métricas seleccionadas para cuantificar la eficacia de los modelos en un contexto de clasificación multiclas.

2.6.1. Entorno de hardware y software

Todos los modelos y experimentos fueron implementados utilizando el lenguaje de programación Python 3.9.10. La infraestructura de software se basó en un ecosistema híbrido actualizado: se empleó TensorFlow 2.19.0 para las arquitecturas matriciales (CNN) y secuenciales (RNN), mientras que PyTorch 2.6.0 junto con PyTorch Geometric 2.6.1 se utilizaron para el desarrollo de la arquitectura GRU bidireccional y los modelos de grafos (GNN).

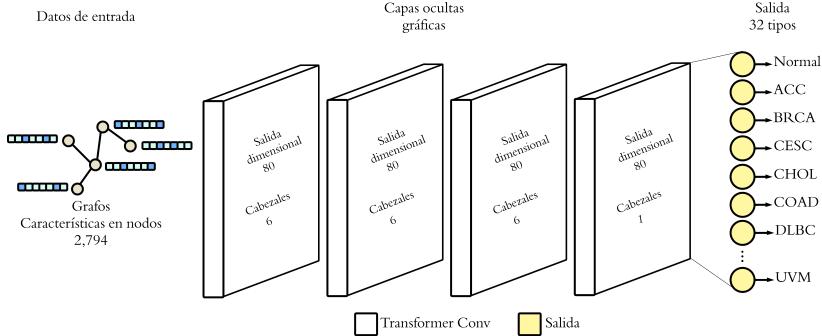


Figure 10: Arquitectura Graph Transformer (GTN). Diagrama del flujo de atención mediante bloques *TransformerConv* (80×6 cabezas). A diferencia de los modelos piramidales, esta red mantiene una densidad constante de 480 dimensiones para el refinamiento de características globales. Se detalla la transición desde la entrada ($N = 2,794$) hasta la salida de 32 clases, integrando Dropout de 0.5 y activaciones LeakyReLU.

El entrenamiento de los modelos se llevó a cabo en un servidor de cómputo de alto rendimiento operando bajo el sistema operativo Ubuntu 24.04.3 LTS. Dado el volumen masivo de los grafos y matrices generados, el procesamiento se realizó en una arquitectura basada en CPU, utilizando un procesador Intel® Xeon® E5-4620 v2 (64 núcleos lógicos) respaldado por una memoria RAM de 320 GB, lo que permitió la carga íntegra de los datasets en memoria y el manejo eficiente de tensores de alta dimensionalidad sin requerir aceleración por hardware (CUDA).

2.6.2. Partición de Datos y Protocolo

El conjunto de datos total se sometió a una partición estratificada del 75/25. El 75% de las muestras se destinó al conjunto de entrenamiento (training set) para el ajuste de los pesos sinápticos, mientras que el 25% restante se reservó estrictamente como conjunto de validación (validation set) para la evaluación final del desempeño. Esta proporción garantiza un volumen suficiente de datos para la generalización del modelo, manteniendo al mismo tiempo un subconjunto de prueba estadísticamente significativo para las 32 clases.

2.6.3. Métricas de evaluación

Dada la naturaleza crítica de la clasificación de tipos de cáncer y el desbalance de clases inherente al conjunto de datos, la evaluación del rendimiento no se limitó a la precisión global. Por consiguiente, se adoptó un enfoque multidimensional alineado con los estándares de la literatura en aprendizaje profundo clínico [29], priorizando métricas que cuantifiquen la robustez de las predicciones en espacios de alta dimensionalidad.

A. Exactitud

Se calculó la exactitud como la proporción global de predicciones correctas. si

bien es una métrica estándar para monitorear la convergencia, su interpretación definitiva se debe realizar junto con otras métricas. En el contexto de aprendizaje profundo con datos desbalanceados, se ha demostrado que esta métrica puede ofrecer una visión incompleta del rendimiento del modelo [30].

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Donde TP (True Positives) representa las instancias correctamente clasificadas como positivas, TN (True Negatives) las correctamente clasificadas como negativas, FP (False Positives) las incorrectamente clasificadas como positivas y FN (False Negatives) las incorrectamente clasificadas como negativas.

B. Precisión y sensibilidad

Para evaluar la calidad el desempeño detallado de cada clase, se calcularon las métricas de precisión (precision) y sensibilidad (recall). La precisión (P) mide la proporción de verdaderos positivos entre todas las predicciones positivas, reflejando la capacidad del modelo para evitar falsos positivos. La sensibilidad (R) cuantifica la proporción de verdaderos positivos detectados entre todos los casos reales positivos, evaluando la habilidad del modelo para identificar correctamente las instancias de cada clase.

$$P = \frac{TP}{TP + FP} , \quad R = \frac{TP}{TP + FN}$$

C. F1-Score

Para armonizar la evaluación entre precisión y sensibilidad, se utilizó el Macro-F1. Esta métrica proporciona una medida balanceada del rendimiento del modelo, especialmente relevante en escenarios de clasificación multiclase con distribución desigual de clases. El enfoque Macro trata a todas las clases con igual importancia estadística ($K = 32$). Esto penaliza severamente a las arquitecturas que logran una alta exactitud global sacrificando el aprendizaje de las clases minoritarias.

$$\text{Macro-F1} = \frac{1}{K} \sum_{i=1}^K \frac{2 \cdot P_i \cdot R_i}{P_i + R_i}$$

Donde P_i y R_i son la precisión y sensibilidad calculadas para la clase i -ésima.

D. Coeficiente de Correlación de Matthews (MCC) multiclase

Se seleccionó el MCC como la métrica definitiva para comparar la generalización de los modelos. En el contexto aprendizaje profundo, el MCC ofrece una validación más robusta de la capacidad predictiva, especialmente en escenarios con clases desbalanceadas. Esta métrica considera todos los elementos de la matriz de confusión, proporcionando una evaluación integral del rendimiento del modelo. La fórmula del MCC para clasificación multiclas es la siguiente [31]:

$$\text{MCC} = \frac{c \cdot s - \sum_k p_k \cdot t_k}{\sqrt{(s^2 - \sum_k p_k^2)(s^2 - \sum_k t_k^2)}}$$

Donde c es el número de muestras correctamente clasificadas, s es el total de muestras, p_k es el número de predicciones para la clase k , y t_k es el número real de instancias de la clase k . El MCC varía entre -1 y +1, donde +1 indica una predicción perfecta, 0 representa una predicción aleatoria y -1 denota una clasificación completamente incorrecta.

3. Resultados

- Rendimiento comparativo con tablas para mostrar las métricas de cada modelo y el tiempo de entrenamiento de cada uno
- Graficos como precision y perdida en entrenamiento y validacion y matrices de confusión

4. Discusión

5. Conclusiones

Acknowledgements

Appendix A. Appendix title 1

Appendix B. Appendix title 2

References

- [1] Organización Mundial de la Salud, Cáncer (2023).
URL <https://www.who.int/news-room/fact-sheets/detail/cancer>
- [2] National Cancer Institute, Cancer statistics (2025).
URL <https://www.cancer.gov/about-cancer/understanding/statistics>
- [3] R. L. Siegel, K. D. Miller, N. S. Wagle, A. Jemal, Cancer statistics, 2023., CA: a cancer journal for clinicians 73 (1) (2023).

- [4] D. S. Dizon, A. H. Kamal, *Cancer statistics 2024: All hands on deck.*, CA: a cancer journal for clinicians 74 (1) (2024).
- [5] D. Hanahan, R. A. Weinberg, *Hallmarks of cancer: the next generation*, cell 144 (5) (2011) 646–674.
- [6] Y. Zhuang, H. Wang, D. Jiang, Y. Li, L. Feng, C. Tian, M. Pu, X. Wang, J. Zhang, Y. Hu, et al., Multi gene mutation signatures in colorectal cancer patients: predict for the diagnosis, pathological classification, staging and prognosis, *BMC cancer* 21 (1) (2021) 380.
- [7] D. Acharya, A. Mukhopadhyay, *A comprehensive review of machine learning techniques for multi-omics data integration: challenges and applications in precision oncology*, *Briefings in functional genomics* 23 (5) (2024) 549–560.
- [8] M. Massimino, F. Martorana, S. Stella, S. R. Vitale, C. Tomarchio, L. Manzella, P. Vigneri, *Single-cell analysis in the omics era: technologies and applications in cancer*, *Genes* 14 (7) (2023) 1330.
- [9] E. Hernández-Lemus, S. Ochoa, *Methods for multi-omic data integration in cancer research*, *Frontiers in Genetics* 15 (2024) 1425456.
- [10] A. Kutlay, Y. Aydin Son, *Integrative predictive modeling of metastasis in melanoma cancer based on microrna, mrna, and dna methylation data*, *Frontiers in Molecular Biosciences* 8 (2021) 637355.
- [11] M. W. Libbrecht, W. S. Noble, *Machine learning applications in genetics and genomics*, *Nature Reviews Genetics* 16 (6) (2015) 321–332.
- [12] Y. LeCun, Y. Bengio, G. Hinton, *Deep learning*, *nature* 521 (7553) (2015) 436–444.
- [13] F. Sartori, F. Codicè, I. Caranzano, C. Rollo, G. Birolo, P. Fariselli, C. Pantocetti, *A comprehensive review of deep learning applications with multi-omics data in cancer research*, *Genes* 16 (6) (2025) 648.
- [14] T. Ye, S. Li, Y. Zhang, *Genomic pan-cancer classification using image-based deep learning*, *Computational and Structural Biotechnology Journal* 19 (2021) 835–846.
- [15] Y.-H. Chuang, S.-H. Huang, T.-M. Hung, X.-Y. Lin, J.-Y. Lee, W.-S. Lai, J.-M. Yang, *Convolutional neural network for human cancer types prediction by integrating protein interaction networks and omics data*, *Scientific reports* 11 (1) (2021) 20691.
- [16] R. Parthasarathy, A. Bhowmik, *A novel recurrent neural network framework for prediction and treatment of oncogenic mutation progression*, *arXiv preprint arXiv:2509.12732* (2025).

- [17] L. Barbadilla-Martínez, N. Klaassen, B. van Steensel, J. de Ridder, Predicting gene expression from dna sequence using deep learning models, *Nature Reviews Genetics* (2025) 1–15.
- [18] B. Jiang, Z. Zhang, D. Lin, J. Tang, B. Luo, Semi-supervised learning with graph learning-convolutional networks, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 11313–11320.
- [19] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, et al., Graph attention networks, *stat* 1050 (20) (2017) 10–48550.
- [20] J. N. Weinstein, E. A. Collisson, G. B. Mills, K. R. Shaw, B. A. Ozenberger, K. Ellrott, I. Shmulevich, C. Sander, J. M. Stuart, The cancer genome atlas pan-cancer analysis project, *Nature genetics* 45 (10) (2013) 1113–1120.
- [21] A. Colaprico, T. C. Silva, C. Olsen, L. Garofano, C. Cava, D. Garolini, T. S. Sabedot, T. M. Malta, S. M. Pagnotta, I. Castiglioni, et al., Tcgabiolinks: an r/bioconductor package for integrative analysis of tcga data, *Nucleic acids research* 44 (8) (2016) e71–e71.
- [22] F. Alharbi, A. Vakanski, B. Zhang, M. K. Elbashir, M. Mohammed, Comparative analysis of multi-omics integration using graph neural networks for cancer classification, *IEEE Access* (2025).
- [23] M. D. Robinson, D. J. McCarthy, G. K. Smyth, edger: a bioconductor package for differential expression analysis of digital gene expression data, *bioinformatics* 26 (1) (2010) 139–140.
- [24] M. E. Ritchie, B. Phipson, D. Wu, Y. Hu, C. W. Law, W. Shi, G. K. Smyth, limma powers differential expression analyses for rna-sequencing and microarray studies, *Nucleic acids research* 43 (7) (2015) e47–e47.
- [25] R. Tibshirani, Regression shrinkage and selection via the lasso, *Journal of the Royal Statistical Society Series B: Statistical Methodology* 58 (1) (1996) 267–288.
- [26] S. Jeon, H. R. Jun, J.-Y. Lee, C. O. Sung, S.-M. Chun, Investigating mirna-driven dna methylation: Statistical evidence of gene-specific modulation, *Science Progress* 108 (3) (2025) 00368504251370988.
- [27] M. Fabbri, G. A. Calin, Epigenetics and mirnas in human cancer, *Advances in genetics* 70 (2010) 87–99.
- [28] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proceedings of the IEEE* 86 (11) (2002) 2278–2324.
- [29] M. Grandini, E. Bagli, G. Visani, Metrics for multi-class classification: an overview, *arXiv preprint arXiv:2008.05756* (2020).

- [30] J. M. Johnson, T. M. Khoshgoftaar, Survey on deep learning with class imbalance, *Journal of big data* 6 (1) (2019) 1–54.
- [31] J. Gorodkin, Comparing two k-category assignments by a k-category correlation coefficient, *Computational biology and chemistry* 28 (5-6) (2004) 367–374.