UNIVERSIDAD PERUANA
CAYETANO HEREDIA

Optimized U-Net for Brain Tumor Segmentation

Aplicaciones Clínicas en Señales e Imágenes

Grupo 1

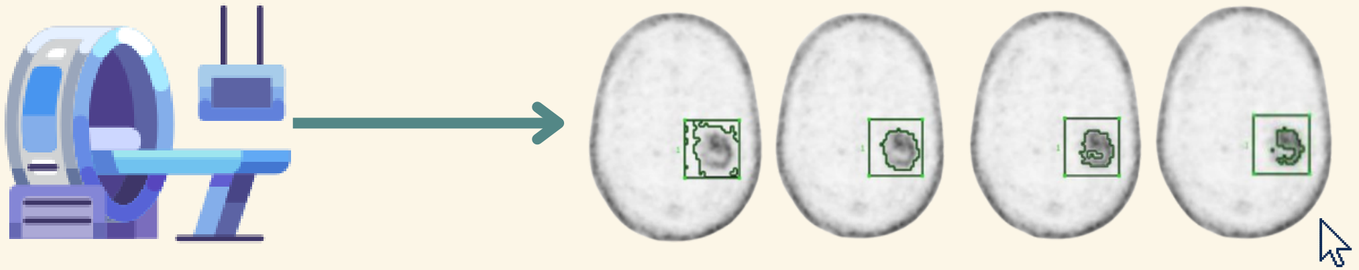
Semestre 2024-2

Contenido

Avance 1 | Entendimiento de la data

Introducción

La segmentación tumoral...



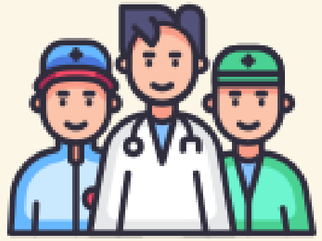
- Identificar
- Delinear regiones afectadas por un tumor

La segmentación automática

- Proporciona asistencia a los radiólogos
- Enfoque más preciso y fiable
- Permite planificación de tratamiento y seguimiento con más tiempo

Problemática

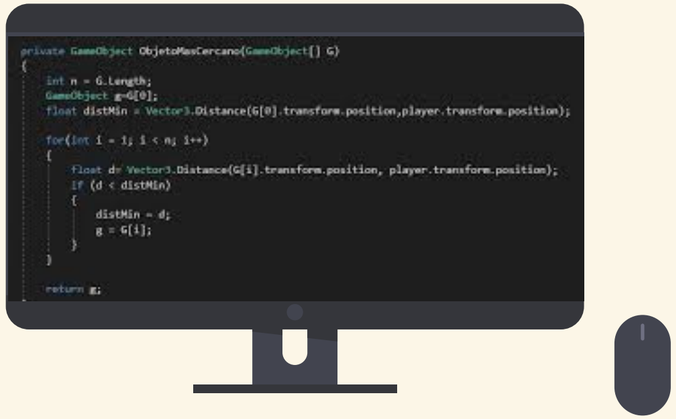
Segmentación manual



Realizada por radiólogos con experiencia

- Proceso que consume mucho tiempo
- Carencia de consistencia y reproducibilidad
- Sujeto a errores humanos

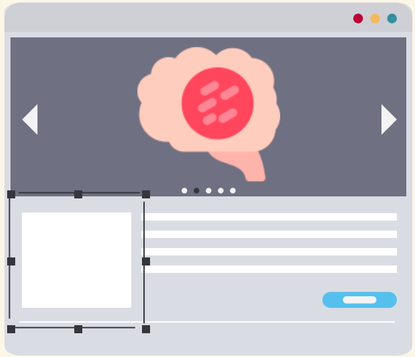
Algoritmos de aprendizaje profundo



```
private GameObject ObjetoMasCercano(GameObject[] G)
{
    int n = G.Length;
    GameObject g=G[0];
    float distMin = Vector3.Distance(G[0].transform.position,player.transform.position);

    for(int i = 1; i < n; i++)
    {
        float d= Vector3.Distance(G[i].transform.position, player.transform.position);
        if (d < distMin)
        {
            distMin = d;
            g = G[i];
        }
    }

    return g;
```

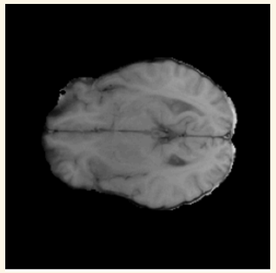


Desafío en diseñar una arquitectura de red neuronal óptima y un programa de enteramiento adecuado

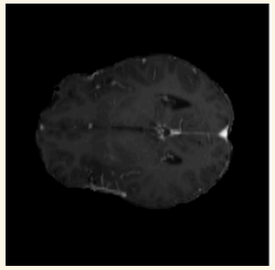
Base de Datos

Se utiliza la base de datos proporcionada por el desafío BraTS21 (Brain Tumor Segmentation Challenge 2021)

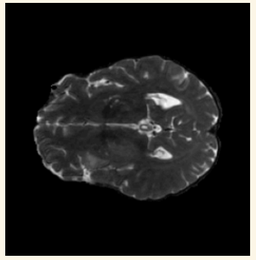
Tipos de imágenes: Imágenes de MRI



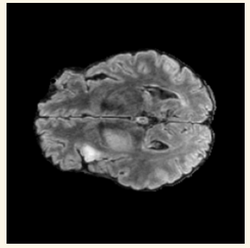
T1: Proporcionan un contraste claro entre la materia gris y blanca



T1Gd: Realzan las áreas con vascularidad de los tumores



T2: Destacan el líquido cerebroespinal y permiten la visualización de edema



FLAIR: Suprime el líquido cerebroespinal para resaltar las lesiones cerebrales

Característica de las imágenes: Almacenadas en *formato NIfTI (.nii)*. Este formato permite manejar volúmenes de datos 3D, preservando la información espacial necesaria para la segmentación precisa.

Características del Conjunto de Datos Resolución:

- Preprocesadas con resolución isotrópica de 1mm3, con dimensiones de 240x240x155 voxeles.
- Etiquetas de Segmentación: Las etiquetas incluyen cuatro clases: tumor realzado (ET), tejido edematoso peritumoral (ED), núcleo necrótico del tumor (NCR), y fondo (voxeles que no son parte del tumor).

Anotaciones

Las anotaciones de los tumores han sido realizadas manualmente **por entre uno y cuatro expertos**, lo que asegura una alta calidad en las etiquetas de segmentación.

Conjunto de datos

Entrenamiento: 1,251 casos
Validación: 219 casos
Prueba 570 casos

Metodología

Preprocesamiento

Cada ejemplo del conjunto de datos BraTS21 consta de cuatro archivos NIfTI con diferentes modalidades de MRI. Se procedio a apilar las cuatros modalidades de manera que cada ejemplo tenga una forma de (4, 240, 240, 155), de manera que el tensor de entrada tenga el siguiente diseño:

(C, H ,W,D) = C-canales, H-altura, W-ancho, D-profundidad

Luego se recortan los voxeles con fondos redundantes (con valor de voxel cero) en los bordes de cada volumen, ya que no brindan información útil y la red neuronal puede ignorarlos. Se calculó la media y desviación estándar dentro de la región distinta de cero para cada canal por separado. Todos los volumenes se normalizaron restando la media y luego diviendo por la desviación estándar. Los voxeles de fondo no se normalizaron, ya que su valor permanecio en cero. Para diferenciar los voxeles normalizados con valor cercano a cero, con los voxeles de fondo, se creó un canal adicional con codificacion one-hot para los voxeles de primer plano y se apilo con los datos de entrada.

Metodología

Ampliacion de datos

La ampliación de datos alivia el problema de sobreajuste al extender artificialmente un conjunto de datos durante el entrenamiento. Para que el modelo sea más robusto, se utilizaron las siguientes ampliaciones:

- Recorte sesgado: Del volumen de entrada, se recorto aleatoriamente un parche de dimensiones (5, 128, 128, 128). Además, con una probabilidad de 0.4, se garantiza que el parche seleccionado mediante recorte sesgado aleatorio tenga algunos vóxeles de primer plano (con clase positiva en la verdad fundamental) en la región recortada.

- Zoom: Con una probabilidad de 0.15, se muestra un valor aleatorio de manero uniforme de (1.0, 1.4) y el tamaño de la imagen se redimensiona a su tamaño original multiplicado por el valor muestreado con la interpolación cúbica, mientras que la verdad fundamental se mantiene con la interpolación del vecino más cercano.

- Flips: Con probabilidad de 0.5, para cada eje x, y, z independiente, volumen fue volteado a lo largo de ese eje.

Metodología

Ampliacion de datos

- Ruido gaussiano: Con una probabilidad de 0.15, se muestrea ruido gaussiano aleatorio con media cero y desviacion estandar muestreada uniformemente desde (0, 0.33), para cada voxel y se agrega al volumen de entrada.

- Desenfoque gaussiano: Con una probalidad de 0.15, se aplica al volumen de entrada un desenfoque gaussiano con una desviación estándar del núcleo gaussiano muestreado uniformemente de (0.5, 1.5).

- Brillo: Con una probabilidad de 0.15, se muestrea un valor aleatorio de manera uniforme de (0.7, 1.3) y luego los voxeles del volumen se multiplican por el.

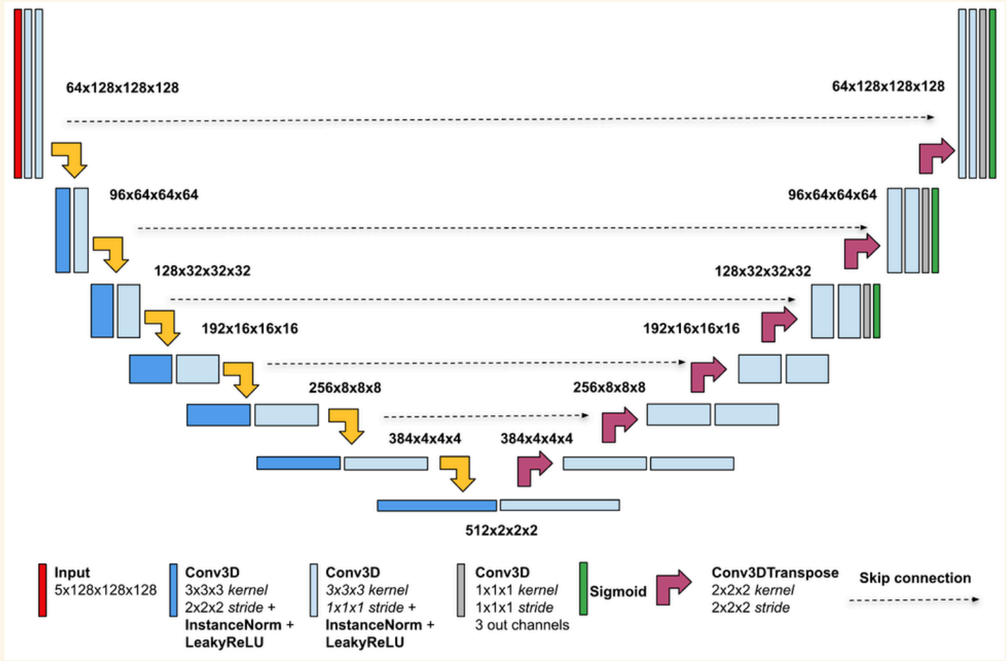
- Contraste: Con una probabilidad de 0.15, se muestrea un valor aleatorio de manera uniforme de (0.65, 1.5) y luego los voxeles de volumen de entrada se multiplican por el y se recortan a su rango de valor original.

Metodología

Arquitectura de modelo: U-Net

Es una red neuronal diseñada específicamente para tareas de segmentación de imágenes, especialmente en el ámbito médico. Su nombre se debe a su forma distintiva en forma de "U". Se divide en dos partes:

- Codificador: Esta parte de la red se encarga de reducir el tamaño de la imagen original, extrayendo las características más relevantes. Lo hace a través de una serie de capas convolucionales que aplican filtros a la imagen para detectar patrones.
- Decodificador: A partir de la información comprimida del codificador, el decodificador reconstruye una imagen de salida con la misma dimensión que la original, pero ahora cada píxel está clasificado según la categoría a la que pertenece (por ejemplo, tumor, tejido sano, etc.). Para lograr esto, utiliza capas de convolución transpuesta que aumentan el tamaño de la imagen y combinan la información con las características extraídas en el codificador.

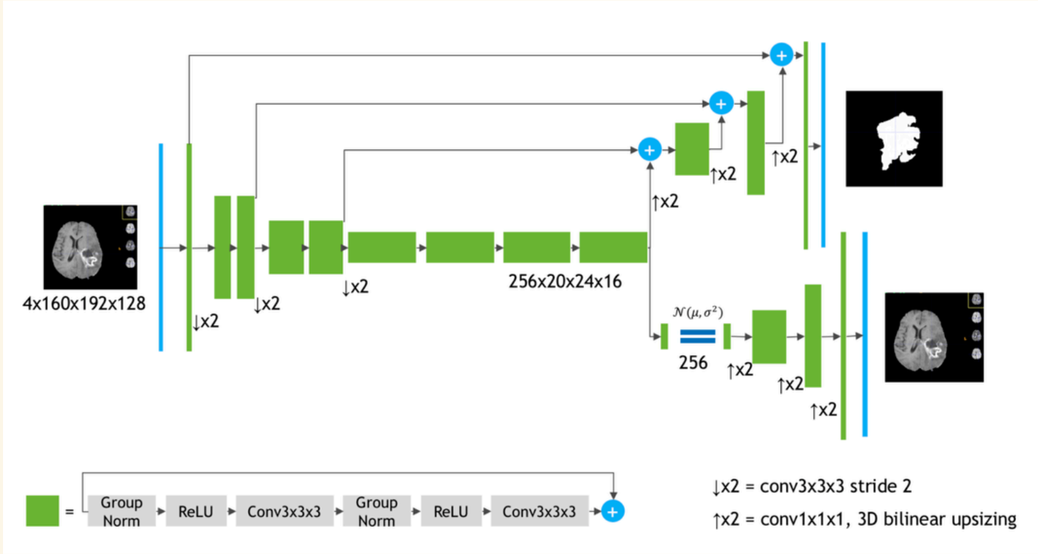


Metodología

Arquitectura de modelo: SegResNetVAE

Es una mejora de la arquitectura U-Net, diseñada específicamente para tareas de segmentación de imágenes, especialmente en el ámbito médico. Se ha destacado por ganar el desafío BraTS 2018. El codificador es similar al U-Net original, pero utiliza bloques residuales para extraer características. El decodificador también es similar al U-Net, pero con una rama adicional de VAE. Esta rama reduce las características del cuello de botella a un espacio de baja dimensión, luego muestrea de una distribución gaussiana y reconstruye la imagen original.

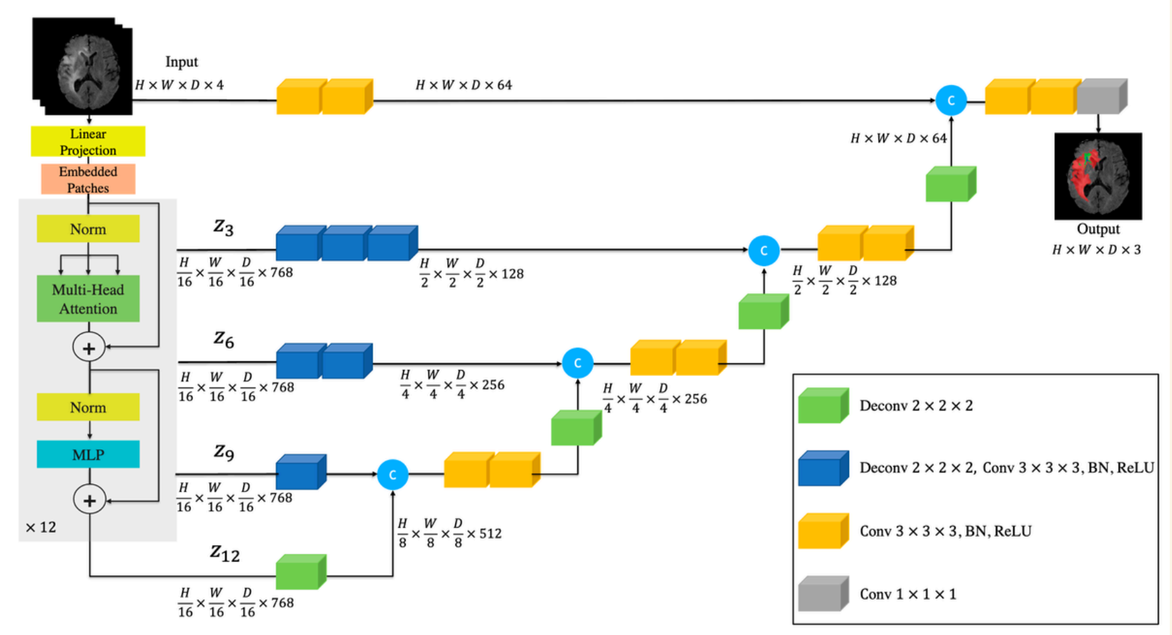
- Bloques residuales: Utiliza bloques residuales similares a los de ResNet, lo que ayuda a prevenir el problema del gradiente desvaneciente y mejora el entrenamiento.
- Variational Autoencoder (VAE): Incorpora una rama de VAE en el decodificador, lo que introduce una regularización y ayuda a reconstruir la imagen original.



Metodología

Arquitectura de modelo: UNETR

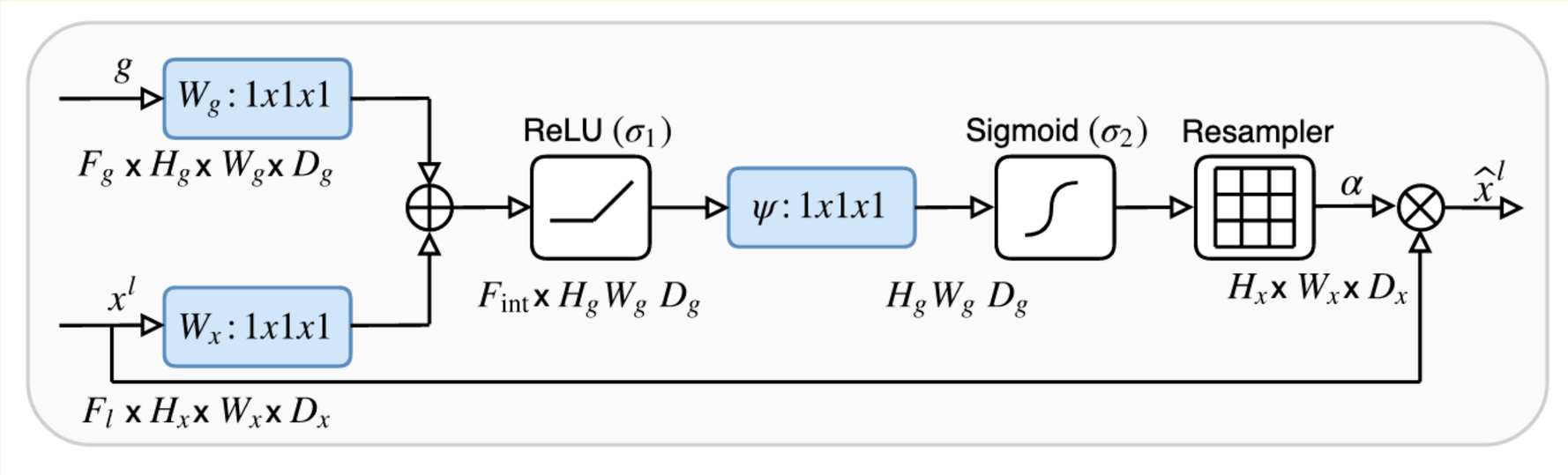
Es una generalización de Vision Transformer (ViT) a las convoluciones 3D: reemplaza las convoluciones 3D en el codificador con autoatención multicabezal. Para convertir un volumen de entrada 3D en una entrada para una autoatención multicabezal, se divide en una secuencia de parches uniformes no superpuestos (con forma de 16x16x16) y se proyecta en un espacio de incrustación (con 768 dimensiones) utilizando una capa lineal, y se le agrega una incrustación posicional. Luego, dicha entrada se transforma mediante un codificador de autoatención multicabezal.



Metodología

Arquitectura de modelo: Attention U-Net

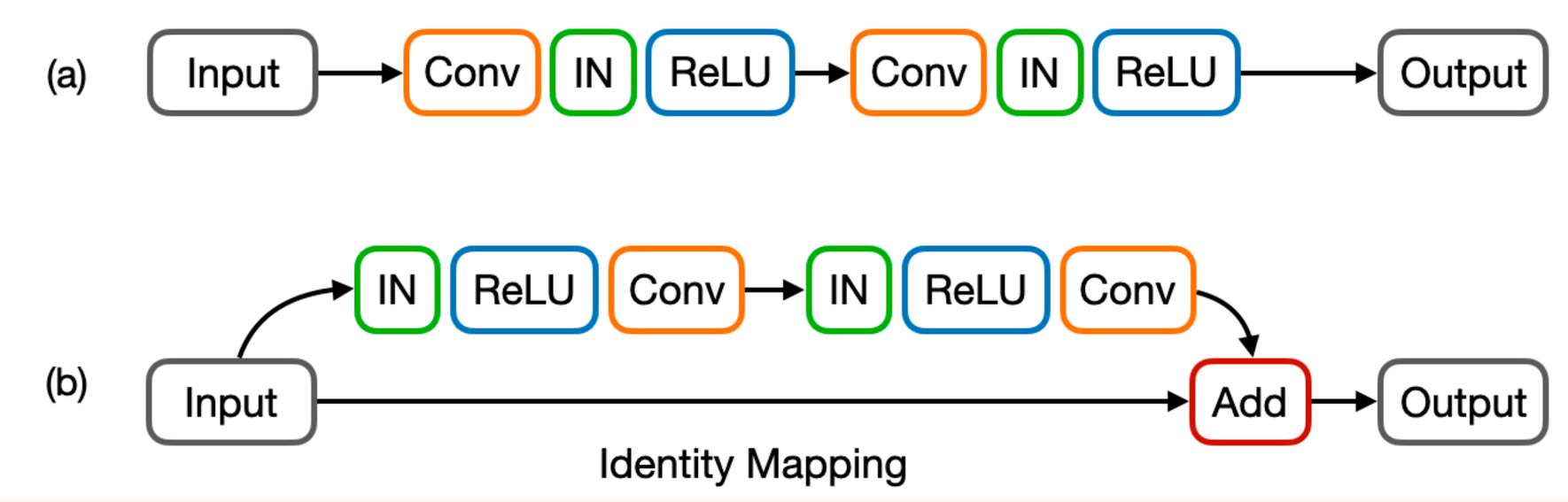
Extiende la base U-Net añadiendo una compuerta de atención en la parte del decodificador. La compuerta de atención transforma el mapa de características del codificador antes de la concatenación en el bloque del decodificador. Aprende qué regiones del mapa de características del codificador son las más importantes, considerando el contexto del mapa de características del bloque del decodificador anterior. Esto se logra mediante la multiplicación del mapa de características del codificador con los pesos calculados por la puerta de atención. Los valores de peso están en el rango (0, 1) y representan el nivel de atención que la red neuronal le presta a un píxel determinado.



Metodología

Arquitectura de modelo: Residual U-Net

Está inspirada en un modelo ResNet en el que se propusieron conexiones residuales. La incorporación de conexiones residuales ayuda a entrenar una red neuronal profunda debido a un mejor flujo de gradiente. La única diferencia entre la red U básica y la red U residual es el cálculo dentro de un bloque convolucional.



Metodología

Arquitectura de modelo

Despues de evaluar cada modelo visto anteriormente, se vio que una U-Net básica logra los mejores resultados y fue seleccionada para una mayor exploración. La siguiente optimización fue ajustar la profundidad del codificador y la selección óptima de los canales de convolución. Como línea base, se utilizó una arquitectura U-Net predeterminada del marco nnU-Net, es decir, la profundidad de la red era 6 y los canales de convolución en cada nivel del codificador eran: 32, 64, 128, 256, 320, 320. Nuestros experimentos han demostrado que aumentar la profundidad del codificador a 7 y modificar el número de canales a: 64, 96, 128, 192, 256, 384, 512 mejora aún más la puntuación de la línea base.

Metodología

Funcion de perdida

Las clases presentes en la etiqueta se convirtieron en las tres regiones parcialmente superpuestas: tumor completo (WT) que representa las clases 1, 2, 4; núcleo tumoral (TC) que representa las clases 1, 4; y tumor enriquecido (ET) que representa la clase 4. Es beneficioso construir la función de pérdida en función de las clases utilizadas para el cálculo de la clasificación, por lo que se diseñó el mapa de características de salida para que tenga tres canales (uno por clase) que, al final, se transforman a través de la activación sigmoidea. Cada región se optimizó por separado con una suma de entropía cruzada binaria o pérdida focal (con el parámetro gamma establecido en 2) con la pérdida de Dice. Para la pérdida de Dice, se utilizó su variante por lotes, es decir, la pérdida de Dice se calculó sobre todas las muestras del lote en lugar de promediar la pérdida de Dice sobre cada muestra por separado.

Metodología

Inferencia

Durante la inferencia, el volumen de entrada puede tener un tamaño arbitrario, en lugar del tamaño fijo del parche (128, 128, 128) como durante la fase de entrenamiento.

Técnicas de inferencia:

- Sliding window: Para procesar imágenes de cualquier tamaño, se utiliza una ventana deslizante que recorre la imagen, y las predicciones en las regiones superpuestas se promedian. Donde la ventana tiene el mismo tamaño que el parche de entrenamiento, es decir, (128, 128, 128) y las ventanas adyacentes se superponen en la mitad del tamaño de un parche. Luego, las predicciones sobre las regiones superpuestas se promedian con ponderación de importancia gaussiana, de modo que los pesos de los vóxeles centrales tienen mayor importancia.

- Test time augmentations: Durante la inferencia, se ha creado ocho versiones del volumen de entrada, de modo que cada versión corresponde a uno de los ocho posibles cambios a lo largo de la combinación de los ejes x, y, z. Luego, ejecutamos la inferencia para cada versión del volumen de entrada y transformamos las predicciones de nuevo a la orientación original del volumen de entrada aplicando los mismos cambios a las predicciones que se utilizaron para el volumen de entrada. Por último, se promediaron las probabilidades de todas las predicciones.

Metodología

Inferencia

- Al optimizar las tres regiones superpuestas (ET, TC, WT), tuvimos que convertirlas nuevamente a las clases originales (NCR, ED, ET). La estrategia para transformar las clases nuevamente a la original es la siguiente: si la probabilidad WT para un vóxel dado es menor que 0.45, entonces su clase se establece en 0 (fondo); de lo contrario, si la probabilidad para TC es menor que 0.4, la clase del vóxel es 2 (ED) y, finalmente, si la probabilidad para ET es menor que 0.45, el vóxel tiene clase 1 (NCR) o, de lo contrario, 4 (ET).
- Post procesamiento: Se aplico post procesamiento para encontrar componentes conectados a ET, para componentes menores a 16 vóxeles con probabilidad media menor a 0.9, reemplazar su clase a NCR (de modo que los vóxeles aún se consideren parte del núcleo del tumor), luego, si hay en general menos de 73 vóxeles con ET y su probabilidad media es menor a 0.9, reemplazar todos los vóxeles de ET a NCR. Con dicho posprocesamiento evitamos el caso límite donde el modelo predijo algunos vóxeles con tumor realzado, pero no había ninguno en la verdad fundamental. Dicho posprocesamiento fue beneficioso para la puntuación final, ya que si no había vóxeles de tumor realzado en la etiqueta, entonces la puntuación Dice para predicción de cero falsos positivos era 1 y 0 en caso contrario.

Metodología

Esta metodología se probó en los conjuntos de validaciones de la validación cruzada de cinco pasos. Se seleccionaron hiperparámetros para obtener la puntuación más alta combinada en todos los pasos. El valor umbral se seleccionó mediante un método de búsqueda en cuadrícula con un paso de 0.05 en el rango (0.3, 0.7). De manera similar, se buscó el número de vóxeles en el rango (0, 100) y se seleccionó maximizando la puntuación en la validación cruzada de cinco pasos.

Resultados

Implementación

La solución está escrita en PyTorch y extiende la implementación de NVIDIA de nnU-Net. El código está disponible públicamente en el repositorio de GitHub de ejemplos de aprendizaje profundo de NVIDIA. La solución propuesta utiliza el contenedor Docker NVIDIA NGC PyTorch 21.07, que permite la encapsulación completa de dependencias, ejecuciones reproducibles y una implementación sencilla en cualquier sistema. Todas las ejecuciones de entrenamiento e inferencia se realizaron con el uso de precisión mixta, que acelera el modelo y reduce el consumo de memoria de la GPU. Los experimentos se ejecutaron en el sistema NVIDIA DGX A100 (8×A100 80 GB).

Resultados

Programa de entrenamiento

Cada experimento se entrenó durante 1000 épocas utilizando el optimizador Adam con tres tasas de aprendizaje diferentes: 0,0005, 0,0007, 0,0009 y una disminución de peso igual a 0,0001. Además, durante los primeros 1000 pasos, se utilizó un calentamiento lineal de la tasa de aprendizaje, comenzando desde 0 y aumentándola hasta el valor objetivo, y luego se disminuyó con un programador de recocido de coseno. Los pesos para las convoluciones 3D se inicializaron con la inicialización de Kaiming. Para la evaluación del modelo, utilizamos una validación cruzada de 5 pliegues y comparamos el promedio de la puntuación Dice más alta alcanzada en cada uno de los 5 pliegues. La evaluación en el conjunto de validación se ejecutó después de cada época. Para cada pliegue, se almacenó los dos puntos de control con la puntuación Dice media más alta en el conjunto de validación alcanzada durante la fase de entrenamiento. Luego, durante la fase de inferencia, reunimos las predicciones de los puntos de control almacenados promediando las probabilidades.

Resultados

Experimentación

Los resultados presentados en la tabla han demostrado que la U-Net de referencia logra la puntuación más alta. Aunque la puntuación de SegResNetVAE es similar a la de la U-Net simple, el tiempo de entrenamiento es tres veces más largo en comparación con la U-Net, debido a la rama VAE adicional. Por lo tanto, decidimos seleccionar la arquitectura U-Net para una exploración más profunda.

Model	U-Net	UNETR	SegResNetVAE
Fold 0	**0.9087**	0.9044	0.9086
Fold 1	**0.9100**	0.8976	0.9090
Fold 2	**0.9162**	0.9051	0.9140
Fold 3	**0.9238**	0.9111	0.9219
Fold 4	**0.9061**	0.8971	0.9053
Mean Dice	**0.9130**	0.9031	0.9118

Resultados

Experimentación

Se probaron varios ajustes de la arquitectura U-Net: atención del decodificador, supervisión profunda, conexiones residuales y bloqueo de caída. Además, se experimentó con la función de pérdida modificada con pérdida focal en lugar de entropía cruzada, de modo que la función de pérdida fuera Focal+Dice. Los resultados experimentales presentados en la tabla han demostrado que la única extensión que mejora significativamente la puntuación media de Dice de 5 veces sobre la U-Net de referencia (0,9130) fue la supervisión profunda (0,9149).

Model	baseline	Attention	DS	Residual	DB	Focal
Fold 0	0.9087	0.9091	**0.9111**	0.9087	0.9096	0.9094
Fold 1	0.9100	0.9110	**0.9115**	0.9103	0.9114	0.9026
Fold 2	0.9162	0.9157	**0.9175**	**0.9175**	0.9159	0.9146
Fold 3	0.9238	0.9232	**0.9268**	0.9233	0.9241	0.9229
Fold 4	0.9061	0.9061	**0.9074**	0.9070	0.9071	0.9072
Mean Dice	0.9130	0.9130	**0.9149**	0.9134	0.9136	0.9133

Resultados

Experimentación

Los resultados en la siguiente tabla han demostrado que aplicar cada una de las modificaciones por separado mejora ligeramente la puntuación sobre el valor inicial de U-Net con supervisión profunda (0,9149), sin embargo, si se utilizan todas las modificaciones juntas, la puntuación mejora aún más (0,9156).

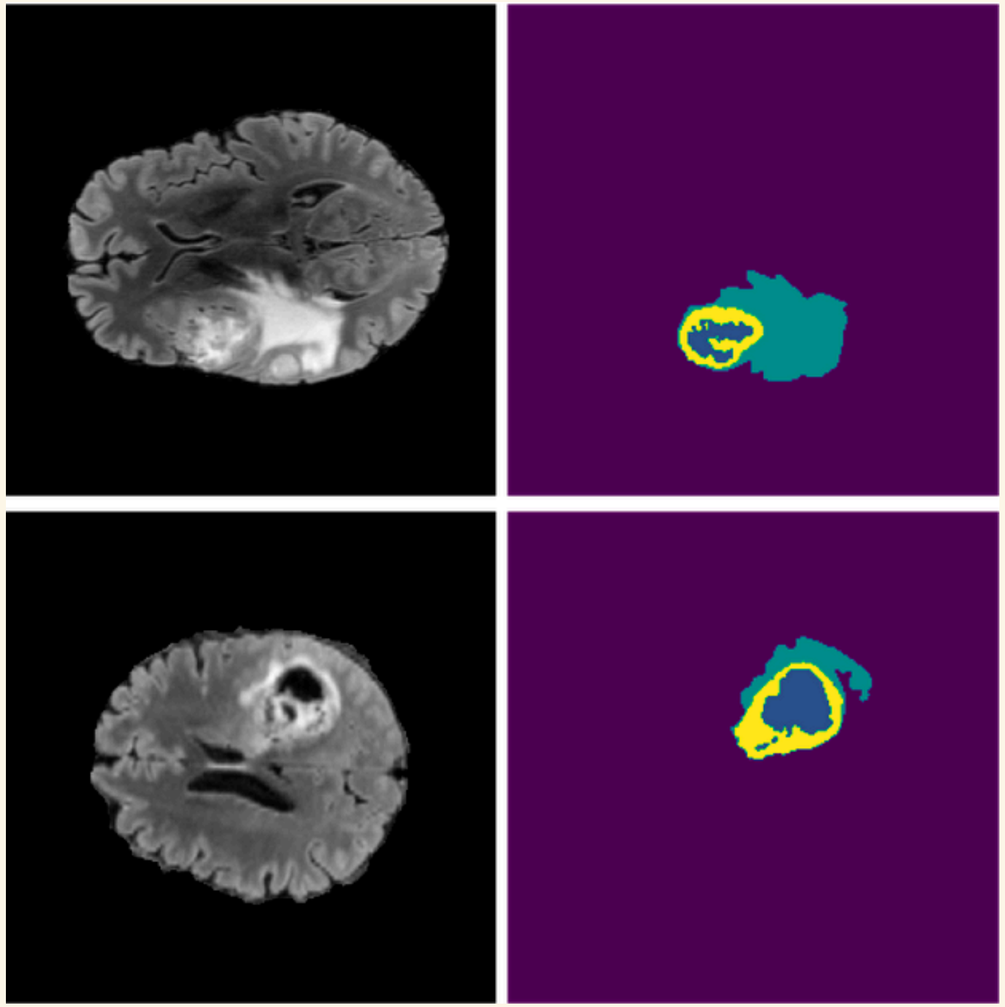
Model	DS	Deeper	Channels	One-hot	D+C+O
Fold 0	0.9111	**0.9118**	0.9107	0.9109	**0.9118**
Fold 1	0.9115	0.9140	0.9135	0.9132	**0.9141**
Fold 2	0.9175	0.9170	0.9173	0.9174	**0.9176**
Fold 3	**0.9268**	0.9256	0.9265	0.9263	**0.9268**
Fold 4	0.9074	**0.9079**	0.9072	0.9075	0.9076
Mean Dice	0.9149	0.9152	0.9150	0.9050	**0.9156**

Puntuaciones Dice promedio de las clases ET, TC, WT para cada 5 pliegues comparando la supervisión profunda (DS), el codificador U-Net más profundo, el número modificado de canales de convolución, el canal de entrada adicional con codificación one-hot para vóxeles de primer plano y todas las modificaciones aplicadas juntas (D+C+O), es decir, U-Net más profundo con un número modificado de canales de convolución y un canal de codificación one-hot para vóxeles de primer plano.

Resultados

Experimentación

Predicciones sobre el conjunto de datos de validación del desafío. En la columna de la izquierda se visualiza la modalidad FLAIR mientras que en la columna de la derecha se visualizan las predicciones del modelo donde el significado de los colores es el siguiente: violeta - fondo, azul - NCR, turquesa - ED, amarillo - ET.



GRACIAS