



Guia de TP REDES NEURONALES
- *Procesamiento Avanzado de Imágenes en Biología y*
Biomedicina-

Profesores: Roberto Sebastián Tomás

Código de la materia: 16.85

Departamento: Bioingeniería

Período Lectivo: 1 de Agosto - 30 de Noviembre

Actividades

Se debe desarrollar un trabajo por grupo en Collab (.ipynb), utilizando el recurso de código y texto para hacer un informe completo del trabajo. El documento debe contener los códigos elaborados por el grupo trabajo y las justificaciones, análisis de los resultados y conclusiones correspondiente para cada actividad que se les propone. El trabajo es grupal y tienen 15 días para entregarlo y deberán hacerlo antes de las 14:00hs del día de la exposición. En esta Guía deberán elegir un dataset para realizar las actividades, podrán optar por los siguientes enlaces:

Kaggle: repositorio de datasets de imágenes médicas.

-Chest X-Ray Images (Pneumonia): <https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia>

-Skin Cancer MNIST: HAM10000 para clasificación de lesiones cutáneas:

<https://www.kaggle.com/kmader/skin-cancer-mnist-ham10000>

Medical Decathlon: datasets de imágenes médicas en diferentes modalidades.

Descargar desde: <https://www.medicaldecathlon.com/>

ISIC Archive: imágenes dermatológicas de alta calidad para el diagnóstico de enfermedades de la piel.

Descargar desde: <https://www.isic-archive.com/>

1- Entrenar una CNN sencilla en un dataset reducido, ajustando los parámetros de entrenamiento: tasa de aprendizaje, tamaño de lote, épocas. Evaluar la precisión y discutir los resultados.

- I. Usar el dataset de imágenes médicas que eligieron para desarrollar la Guía IV. Preprocesar las imágenes (redimensionar, normalizar) para asegurar que el tamaño sea uniforme. Construir una CNN básica con Keras, incluyendo capas de convolución, pooling y capas totalmente conectadas al final. Entrenar la red e interpretar resultados, analizar errores.
- II. Aplicar aumentos de datos (rotación, escalado, etc.) y dropout y volver a entrenar la CNN. Analizar y comparar resultados con el inciso anterior
- III. Utilizar un modelo preentrenado como ResNet o VGG16 con pesos de ImageNet para clasificación. Adaptar el modelo para una tarea de clasificación específica en imágenes biomédicas, como diferenciación de tejidos. Congelar las capas del modelo base y entrenar solo las capas superiores en el dataset biomédico. Interpretar los resultados y hacer un análisis de errores.
- IV. Descongelar algunas de las capas superiores de VGG o ResNet y entrenar nuevamente en el dataset de imágenes médicas. Comparar resultados con el inciso anterior.

Comparar los resultados obtenidos en cada inciso ¿Que conclusiones puede sacar?, ¿Cuál es el tipo de RED que funciona mejor? ¿Cómo haría para evaluar si la calidad de la imagen afecta la clasificación?

2- Entrenar una CNN como U-Net en una tarea de segmentación de imágenes médicas.

- V. Diseñar una red tipo U-Net, teniendo en cuenta que se ajuste al tamaño y tipo de datos de las imágenes médicas. Entrenar el modelo para segmentar áreas específicas en las imágenes. Analizar los resultados y evaluar errores.
- VI. Descargar y cargar un modelo U-Net preentrenado para segmentación. Evaluar el modelo entrenado en la tarea de clasificación o segmentación, interpretar los resultados y hacer un análisis de errores. Comparar con el inciso anterior.
- VII. Ajustar los hiperparámetros del modelo y comparar resultados antes y después de la optimización. Probar diferentes configuraciones en el modelo y analizar el impacto en la precisión y la pérdida.

Comparar los resultados obtenidos en cada inciso ¿Que conclusiones puede sacar?, ¿Cuál es el tipo de U-NET que funciona mejor?¿Como haría para evaluar si la calidad de la imagen afecta la segmentación?

3- Comparar las métricas de precisión, sensibilidad y especificidad entre los modelos (CNN de cero, VGG, ResNet y U-Net). Usar Grad-CAM para interpretar el modelo de clasificación y visualizar qué áreas de las imágenes activaron más la red.

4- Cargar un dataset de imágenes médicas y preprocesarlas si es necesario. Construir un GAN básica desde cero usando TensorFlow o PyTorch.

VIII. Implementar el Generador que toma ruido como entrada y produce imágenes médicas simuladas, y un Discriminador que distinga entre las imágenes reales y las generadas.

Entrenar el modelo combinando el generador y el discriminador, ajustar los parámetros de entrenamiento (tasa de aprendizaje, tamaño de lote, épocas). Visualizar imágenes generadas durante el entrenamiento para evaluar el progreso.

IX. Implementar una DCGAN para mejorar la calidad y control de la generación. Evaluar las imágenes generadas.

X. Cargar un modelo ViT preentrenado y adaptar las capas finales para una tarea de clasificación específica. Comparar la precisión y el rendimiento entre los modelos CNN y ViT en el dataset de imágenes médicas. Aplicar Grad-CAM o visualizaciones de autoatención para entender qué áreas de la imagen activan más el modelo.

Comparar precisión, F1-score, y otras métricas relevantes entre los modelos ViT y GANs. En el caso de GANs, evaluar la calidad visual de las imágenes generadas usando métricas de similitud estructural (SSIM).