INFORME ROYECTO

Docente: RAUL RAMOS POLLAN

Estudiante: ELKIN DAVID SANCHEZ VELEZ

Fecha: 11/2024

UNIVERSIDAD DE ANTIOQUIA

Facultad de Ingeniería

Fundamentos de Deep Learning

Introducción

Las cianobacterias, comúnmente conocidas como algas **verde-azuladas**, son microorganismos fotosintéticos fundamentales en los ecosistemas acuáticos de agua dulce. Aunque desempeñan un papel crucial en el equilibrio ecológico y la producción de oxígeno, algunas especies representan un riesgo significativo para la salud humana y animal debido a su capacidad de producir cianotoxinas altamente peligrosas.

El proyecto de clasificación de cianobacterias surge como una respuesta tecnológica innovadora a los desafíos de identificación y monitoreo de estas especies en entornos naturales. La identificación precisa y rápida de diferentes especies de cianobacterias es esencial para:

1. **Prevenir riesgos para la salud pública**
2. **Proteger ecosistemas acuáticos**
3. **Desarrollar estrategias de gestión ambiental efectivas**

El conjunto de datos utilizado, **Cyanotoxins Identification Dataset**, compila imágenes de 13 especies diferentes de cianobacterias, proporcionando un recurso fundamental para el desarrollo de herramientas de inteligencia artificial capaces de identificar y clasificar estos microorganismos con precisión.

Nuestro objetivo principal es desarrollar un modelo de aprendizaje profundo que pueda clasificar automáticamente estas especies con alta precisión, utilizando técnicas avanzadas de procesamiento de imágenes y redes neuronales convolucionales.

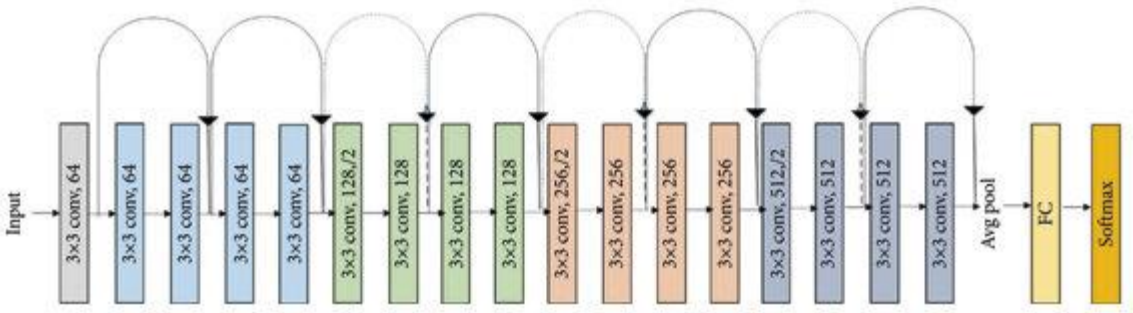
1. Descripción de la estructura del notebook

- **Cargar datos desde Kaggle:** Descarga y descompresión del dataset desde Kaggle, asegurando la accesibilidad de los datos para el análisis.
- **Limpieza de datos:** Aplicación de la función check_image_quality para identificar y mover imágenes con problemas de calidad (dimensiones, desenfoque, contraste, etc.). Resumen de estadísticas por clase y problemas encontrados.
- **Generadores personalizados:** Implementación de un generador balanceado de imágenes para batches en entrenamiento y validación, asegurando una representación equitativa de todas las clases.

- **Creación del modelo:**
    - Definición de la arquitectura ResNet18 con bloques residuales.
    - Configuración de hiperparámetros clave: tamaño de entrada, número de clases, y diseño para facilitar la propagación del gradiente.
- **Entrenamiento del modelo:**
    - Uso de callbacks como parada temprana y reducción de la tasa de aprendizaje.
    - Aumento de datos durante el entrenamiento (rotaciones y translaciones aleatorias).
- **Evaluación del modelo:** Generación de predicciones en el conjunto de datos de prueba y análisis de desempeño con métricas clave.

2. Descripción de la solución

Arquitectura del modelo



- Se implementó una variante personalizada de ResNet18 para clasificación multiclase.
- **Componentes principales:**
    - Bloques residuales: conexiones para facilitar el flujo del gradiente.
    - Normalización por lotes: estabiliza el entrenamiento.
    - Activaciones ReLU: introduce no linealidad.
    - Capa de clasificación final: softmax con Global Average Pooling.
- **Aumento de datos incorporado:**
    - RandomRotation y RandomTranslation: para mejorar la generalización.

Preprocesamiento de imágenes

- Aplicación de verificaciones de calidad para filtrar imágenes problemáticas (e.g., desenfoque, contraste insuficiente, fondo uniforme). Esto ayudó a evitar que datos de baja calidad afectaran el entrenamiento.
- Transformación de las imágenes:
    - Redimensionamiento a 224x224 píxeles.
    - Asegurarse de tener lo datos en RGB
    - Normalización de píxeles al rango [0, 1].

Generador de datos balanceado

- Diseñado para garantizar un **número igual de imágenes por clase** en cada batch entregado al modelo, evitando el sesgo hacia clases con mayor cantidad de datos.
- Integra oversampling (sobremuestreo) para clases menos representadas, aprovechando aumentos de datos como rotaciones y translaciones.
- Funciona en armonía con el aumento de datos para lograr un balance artificial que potencia la eficacia del entrenamiento.

Hiperparámetros principales

- Tasa de aprendizaje inicial: 1 × 10^-5.
- Batch size ajustado dinámicamente para garantizar la divisibilidad entre clases.
- Callbacks:
    - **Reducción de tasa de aprendizaje personalizada:** Disminuye la tasa cada 10 épocas, ajustada tras pruebas empíricas para optimizar convergencia.
    - **Early Stopping:** Detiene el entrenamiento y devuelve los mejores pesos al detectar sobreajuste en el conjunto de validación.

3. Descripción de las iteraciones

Iteración inicial

- Se entrenó el modelo sin un filtrado riguroso de calidad en las imágenes.
- Las primeras épocas mostraron baja precisión debido a imágenes problemáticas y desbalance entre clases.

Iteraciones posteriores

- Se destacó la importancia de filtrar imágenes con fondos sólidos y otros problemas de calidad.
- Ajustes realizados:
    - Implementación del generador balanceado para corregir el desbalance entre clases.
    - Uso de aumentos de datos (rotaciones y translaciones aleatorias) que, junto con el generador, lograron trabajar en armonía para mejorar la representatividad de clases en los batches.
    - Introducción de los callbacks mencionados para optimizar el entrenamiento:
        - Early Stopping redujo el tiempo de entrenamiento al detenerse cuando la validación dejaba de mejorar.
        - Reducción de la tasa de aprendizaje permitió refinar los pesos en etapas avanzadas.

4. Resultados en el conjunto de pruebas

Precisión global: 80.34% sobre **595** imágenes distribuidas en **13 clases**.

- **Desempeño por clase:** Las clases con mejor desempeño incluyeron **Microcystis** (precisión de **96%**) y **Phormidium** (precisión de **83%**), ambas con cantidades significativas de datos y características distintivas. Clases minoritarias como **Aphanizomenon** y **Raphidiopsis** presentaron menor precisión debido al reducido número de imágenes.
- **Factores limitantes:** El modelo enfrentó desafíos significativos debido a la marcada distribución desigual de datos. En el conjunto de entrenamiento, **Microcystis** predomina con **700 imágenes**, mientras que **Raphidiopsis** cuenta con solo **9 imágenes**. Esta disparidad extrema (una diferencia de casi 78 veces) generó un sesgo considerable, dificultando la generalización

del modelo para clases minoritarias y comprometiendo su capacidad de clasificación precisa en especies con representación mínima.

- **Matriz de confusión:** Reveló confusiones frecuentes entre especies con características morfológicas similares, especialmente en clases con pocas muestras.

Matriz de confusión

(Resumen visual que refleja las confusiones más comunes entre clases cercanas).