M2.875 Deep Learning

Práctica:

Implementación de un algoritmo para la clasificación de enfermedades foliares del árbol del manzano

Contenidos

1. Presentación

A lo largo de las tres partes de la asignatura hemos entrado en contacto con diferentes tipos de algoritmos de aprendizaje automático basado en la utilización de técnicas de optimización de Deep Learning que permiten solucionar varios problemas de regresión y clasificación.

Esta práctica permitirá poner en práctica los conocimientos adquiridos a lo largo de la asignatura en aplicación a la resolución de un problema real.

2. Competencias

En esta actividad se trabajan las siguientes competencias:

- Capacidad para analizar un problema desde el punto de vista del aprendizaje automático.
- Capacidad para analizar un problema en el nivel de abstracción adecuado a cada situación y aplicar las habilidades y conocimientos adquiridos para resolverlos.

3. Definición del problema

Las manzanas son uno de los cultivos de fruta más importantes del mundo. Las enfermedades foliares (de hojas) representan una amenaza importante para la productividad y la calidad generales de los manzanos. El proceso actual para el diagnóstico de enfermedades en los huertos de manzanos se basa en la exploración manual por parte de los humanos, que requiere mucho tiempo.

Aunque los modelos basados en la visión por computador han demostrado ser prometedores para la identificación de enfermedades de plantas, hay algunas limitaciones que hay que abordar. Las grandes variaciones en los síntomas visuales de una sola enfermedad entre diferentes cultivos de manzana son los principales retos para la identificación de enfermedades basadas en la visión por computador. Estas variaciones surgen de las diferencias en entornos naturales y de captura de imágenes, por ejemplo, el color y la morfología de las hojas, la edad de los tejidos infectados, el fondo de la imagen no uniforme y las variaciones de las condiciones de iluminación durante la toma de la imagen entre otros.

El proyecto que se presenta es una versión modificada del reto de la plataforma Kaggle alojado en https://www.kaggle.com/c/plant-pathology-2021-fgvc8. El conjunto de datos de la práctica, etiquetado por expertos atendiendo a la enfermedad foliar de la manzana, está formado por dos conjuntos de imágenes: uno de entrenamiento formado por 9.750 imágenes RGB y otro de test sin clasificar formado por un total de 7.527 imágenes. Para facilitar el preprocesamiento, las imágenes se suministran con una resolución reducida. La clase C0 es de hojas sanas y las clases C1 a C5 de diferentes tipos de enfermedades. Este conjunto de datos refleja escenarios de campo reales, representando fondos no homogéneos de imágenes de hojas tomadas en diferentes etapas de madurez y en diferentes momentos del día bajo diferentes parámetros de exposición de la cámara.

4. Objetivos

El objetivo principal de la práctica es desarrollar modelos basados en el aprendizaje automático para clasificar con precisión la enfermedad que afecta a la hoja de la fotografía analizada.

5. Función de evaluación

La función de evaluación que se utilizará para medir la capacidad predictiva y de generalización de los modelos es la F1-macro-average.

6. Recursos

El conjunto de datos del proyecto se puede encontrar en la dirección:
https://www.kaggle.com/jordidelatorreuoc/kaggle-plant-pathology-2021-modificat

Para resolver el problema podéis utilizar:

- Kaggle (www.kaggle.com): Proporciona hasta 36 horas semanales de acceso a TPUs y GPUs de alto rendimiento de forma gratuita.
- Google Colab (https://colab.research.google.com/). Acceso también a GPUs y TPUs.
- Ordenador personal si dispone de GPU y software adecuado (Tensorflow o Pytorch con aceleración hardware. Se recomienda instalar sobre Anaconda como entorno Python).

7. Guía de evaluación

A continuación presentamos una guía de los puntos que se tendrán en cuenta de cara a valorar el proyecto entregado:

Ejercicio 1 (2 puntos)

Elaborar un análisis exploratorio de los datos del problema que queremos resolver mostrando gráficamente al menos la siguiente información:

- Porcentaje de imágenes de cada clase
- Número total de imágenes para clase
- Muestra de al menos 5 imágenes de cada clase

Ejercicio 2 (1 punto)

Definición de la estrategia de validación. (División del conjunto de entrenamiento en entrenamiento propiamente dicho y conjunto de validación).

Ejercicio 3 (5 puntos)

- Definición de la estrategia de aumento de datos sobre el conjunto de entrenamiento.
- Elegir el tipo de modelo y tamaño de imagen de entrada que se utilizará para la predicción.
- Elegir el tipo de estrategia de entrenamiento que se seguirá (entrenamiento desde cero o fine-tuning de un modelo pre-entrenado). Se recomienda fine-tuning.
- Mostrar las gráficas de entrenamiento donde se pueda ver claramente la evolución de la función de pérdida y de la función de evaluación (F1) tanto para el conjunto de entrenamiento como para el de validación.
- Elegir el modelo con mejor rendimiento (aquel que tiene la máxima F1 sobre el conjunto de validación).

Ejercicio 4 (2 puntos)

- Presentar para el modelo que haya dado mejores resultados de la función de evaluación (F1-macro-average) las siguientes medidas de rendimiento adicionales: matriz de confusión, F1-score, Accuracy, Precision y Recall (todas ellas con media ponderada)
- Calcula las predicciones del conjunto de test proporcionado y guarda en un CSV con dos campos: nombre del archivo de la imagen analizada y clase predicha. El archivo debe tener la forma siguiente:

```
image, label
fichero1.jpg, C0
fichero2.jpg, C5
fichero3.jpg, C2
...
```

8. Entrega

El entregable será un archivo comprimido en formato ZIP con los siguientes documentos:

- **Informe en formato PDF** de entre 10 y 15 páginas de longitud, aproximadamente.
- **Código** utilizado, ya sea en ficheros Jupyter notebook (.ipynb) o Python (.py), y la exportación en HTML.
- **Archivo test.csv** con dos campos: *image* y *label*. Deberá haber una entrada para predicción sobre el conjunto de test. En el primer campo estará el nombre del archivo y en el segundo la clase predicha (C0, C1, C2, C3, C4, C5).

Para el informe se puede usar la siguiente guía:

- Tamaño de letra 11 o 12
- Fuente: Arial o similar
- Interlineado sencillo
- Las capturas de pantalla (por ejemplo, las gráficas de rendimiento) o los fragmentos de código (si se consideran relevantes) deben estar pensados para ilustrar y no para ser protagonistas

El **código fuente** utilizado para todas las etapas de la práctica debe estar correctamente comentado para facilitar su comprensión. Podéis utilizar archivos Python nativos (.py) o basados en Jupyter Notebook (en este caso se debe entregar la versión .ipynb, y la exportación en formato .html).

9. Agradecimientos

- Kaggle Platform (www.kaggle.com)
- Cornell Initiative for Digital Agriculture (CIDA).
- Thapa, Ranjita; Zhang, Kai; Snavely, Noah; Belongie, Serge; Khan, Awais. The Plant Pathology Challenge 2020 data set to classify foliar disease of apples. Applications in Plant Sciences, 8 (9), 2020. https://doi.org/10.1002/aps3.11390