
Visión de computadora con redes neuronales convolucionales

Objetivos

- Emplear TensorFlow y Keras para implementar una red neuronal convolucional capaz de clasificar imágenes.

Procedimiento

En la práctica de esta semana usted tendrá como tarea emplear la combinación de TensorFlow y Keras dentro de Python para resolver un problema de visión de computadora empleando deep learning (con una red de poca profundidad, tal que aún pueda entrenarse sólo con CPU) con base en data previamente seleccionada. Para lograr esto, realice lo siguiente:

1. Descargue de Canvas el archivo `mt3006lab4.zip` y extraiga sus contenidos a una carpeta de su selección. Dentro de la carpeta usted encontrará no sólo la data que deberá emplear para los problemas sino que también un script en Python el cual puede emplear como base para sus soluciones.
2. Emplee como base el script `lab4.py` para resolver el problema de **clasificación de imágenes de letras**.
 - a) Se tiene una colección de 3000 imágenes en blanco y negro, de 28x28 cada una, que describen distintas formas de escribir a mano las letras A, B y C. El código para cargar las mismas ya se encuentra dentro del script provisto en Python y se separan en un set de entrenamiento y de validación. Dentro del código también se encuentra la generación automática de las etiquetas de las imágenes mediante un one-hot encoding: 100, 010 y 001 para las categorías 'A', 'B' y 'C' respectivamente. Finalmente, el código también le muestra una figura con 20 ejemplos del set de entrenamiento, seleccionados de forma aleatoria.
 - b) Construya una red neuronal con: una capa convolucional de 16 kernels de tamaño 5x5 y función de activación ReLU (para esta capa debe especificar las dimensiones de la entrada mediante el atributo `input_shape`), una capa de max pooling de tamaño 2x2 y que se salte 2 filas de la imagen en cada corrida (esto corresponde al atributo de `strides`), una capa que aplane el resultado del pooling a un vector columna y una capa densa de 3 salidas con función de activación *softmax* (versión multivariable de la sigmoide). Revise la documentación de las capas disponibles en Keras y haga una correspondencia con lo que se le pide.

- c) Entrene la red neuronal con stochastic gradient descent y una pérdida de entropía cruzada categórica. Usted tiene como tarea establecer el valor de los hiper-parámetros restantes, métricas de rendimiento y generar figuras y resultados pertinentes para el ajuste y evaluación del modelo.
 - d) Con base en las figuras y resultados obtenidos en el inciso anterior, razone, ¿Qué tan exacto es el modelo entrenado? ¿Son interpretables los kernels aprendidos en la capa convolucional? ¿Qué ocurre si se cambia el tamaño de los kernels tanto para el filtrado como para el pooling? ¿Mejora significativamente el rendimiento de la red si se añade una capa oculta a la red neuronal densa de clasificación (mencione qué tipo de función de activación probó y el número de nodos ocultos)? ¿Cómo puede establecer que el modelo no está presentando overfitting? ¿Puede mejorar el rendimiento de la red añadiendo capas de dropout? Genere las figuras que considere pertinentes para respaldar su razonamiento detrás de las preguntas.
3. Cuando esté satisfecho con sus resultados, presente los mismos al catedrático del laboratorio y responda a las preguntas que se le hagan. Recuerde que entregas tardías representan una penalización del 25 % por semana.