

TAREA CALIFICADA #1 INAR 2024

En esta práctica me encontré con el problema que algunos alumnos más presentaban en clase. No podía unirme y participar en kaggle (adjunto foto). Al no poder realizar la práctica en dicha plataforma y acceder a los documentos de las imágenes para así poder realizar los modelos de entrenamiento, inicié sesión con los datos del teléfono de un familiar y me permitió unirme.

Revisa los términos y condiciones del concurso

Al hacer clic en el botón "Entiendo y acepto" a continuación, acepta estar sujeto a las reglas del concurso para [\[U-tad\] Dogs vs. Cats 2024](#).

REGLAS ESPECÍFICAS DE LA COMPETENCIA

- Aunque se trata de una competencia, ganar el aprendizaje; Aprender Ganar != = !
- No hagas trampa, la única persona a la que engañarás es a ti mismo.
(Basta con decir que el etiquetado a mano de las imágenes de prueba es obviamente una forma de trampa)

TÉRMINOS Y CONDICIONES DEL CONCURSO

- ☒ Acepto las reglas específicas de la competencia anteriores y las Reglas fundamentales de la competencia de Kaggle. Las Reglas Fundamentales de la Competencia Kaggle reemplazan las reglas específicas de la Competencia en caso de conflicto.
- ☐ Entiendo que participar en este Concurso crea una relación directa entre el anfitrión y yo. Kaggle no se hace responsable de los premios, compromisos, contenidos o acciones prometidos por un Anfitrión o Participante.

Se denegó el permiso 'competitions.participate'

Rechazar

Entiendo y acepto

Yo utilizo Google Collabs en el trabajo, por lo que me descargué el archivo JSON para así poder utilizar la API de kaggle en Google Collabs y poder realizar la práctica desde ahí.

He realizado varios entrenamientos para así poder tener una visión general de las mejoras o “pérdidas de tiempo y dinero” en los diferentes entrenamientos.

He adjuntado tanto el archivo en formato .ipynb como en .pdf para poder abrirlos en cualquier ordenador y revisar los modelos. Además, adjunto el archivo kaggle.json con mis credenciales por si hiciera falta para volver a ejecutar los notebooks y comprobar así su correcto funcionamiento.

Todos los modelos han sido entrenados con la GPU L4 salvo un modelo que se probó con la A100 que lo comento en el modelo correspondiente.

CNN

cnn00

Es el modelo dado por los profesores adaptando el código para así poderlo ejecutar en Google Collabs utilizando la API y que todo el material se descargue automáticamente.

Cómo el lógico, el resultado del entrenamiento es prácticamente el mismo.

cnn01

Primer modelo realizado partiendo de la base dada por el profesorado. El tamaño de imagen lo dejamos igual (256x256), ajustamos el tamaño del batch a 32, siendo 125 el del modelo base.

Añado varias capas más y dejando 25 épocas de entrenamiento.

Realizando estas mejoras ya obtuvimos un accuracy de 0.7032, siendo este mejor que el proporcionado.

cnn02

En este modelo, parto del anterior y decido únicamente añadir capas convolucionales de más tamaño y más número de ellas para ver si realizando estos cambios en el modelo y añadiendo el doble de épocas, el accuracy mejoraba.

Este modelo ya tardó bastante más en entrenar. El cnn02 tardó unos 10-15min en entrenarse, en cambio, este modelo, al tener más capas y más épocas, tardó 4h y 15min.

Finalmente, el accuracy obtenido fue 0.7154, un dato que comparándolo con el modelo anterior deja bastante claro que este último es un modelo deficiente.

cnn03

El último que propongo lo hago bajando el batch a 16, dejando el mismo números de capas pero estás de mayor tamaño de kernel y metiendo 100 épocas para el entrenamiento.

Este modelo estuvo entrenando 10h y 35min. El resultado del accuracy fue 0.7678, dato notablemente más elevado que los modelos anteriores, mucho más eficiente que el modelo cnn02.

Por último, aclarar que en los modelos CNN no incorporo modelos con tamaños de imágenes inferiores a 256x256 ya que en más de 10 modelos diferentes que realicé todos arrojaban un accuracy entre 0.29 y 0.35.

Transfer Learning

transferlearning00

Modelo dado por los profesores adaptando el código para así poderlo ejecutar en Google Collabs utilizando la API y que todo el material se descargará automáticamente.

El resultado obtenido es prácticamente el mismo.

transferlearning01

He cambiado el modelo VGG16 que es el dado por el profesorado por el ResNet50, que es un modelo preentrenado más robusto para la clasificación de imágenes. El tamaño de imagen y demás parámetros como el batch he dejado los mismos. Los resultados de este cambio han sido muy buenos, ya que el ha tardado 3 veces menos en entrenarse y además mejorando levemente el entrenamiento.

transferlearning02

En este caso quise realizar un entrenamiento mucho más complejo. El tamaño de imagen lo aumente a 600x600, el batch size lo reduje a 32, aumenté las épocas a 25 y por último utilicé el modelo EfficientNetB7 que es muy más avanzado. Me devolvió un error, el cual significa que el modelo está consumiendo mucho memoria, probé a entrenar con la gráfica A100 de Google Collabs, la cual otorga 40gb de RAM en GPU y aún así no fue suficiente, por lo que este modelo lo abandoné.

transferlearning03

Viendo los problemas y errores que me arrojó el anterior modelo, decidí realizar un modelo que utilizara menos memoria. Para ello reduje el tamaño de imagen a 380x380, cambié el modelo a uno más ligero (EfficientNetB5). El número de épocas no lo cambié, lo dejé en 25.

Este último modelo tardó en entrenarse 1h y 55min. pero fue el modelo que mejores resultados proporcionó de los distintos modelos ya pre-entrenados.