Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Querétaro

TC3007C. Inteligencia artificial avanzada para la ciencia de datos II
Grupo 501

Momento de Retroalimentación :

"Módulo 2 Implementación de un modelo de deep learning. (Portafolio Implementación)"

Evidencia presentada por los estudiantes :

Emiliano Mendoza Nieto A01706083

Profesores :

Benjamín Valdés Aguirre
José Antonio Cantoral Ceballos
Carlos Alberto Dorantes Dosamantes
Eduardo Daniel Juárez Pineda
Ismael Solís Moreno

Fecha de entrega :

25 de Noviembre de 2023

I. INTRODUCCIÓN

El proyecto tiene como objetivo desarrollar un modelo de clasificación de marcas de automóviles utilizando técnicas de Deep Learning. Esta tarea es relevante en aplicaciones como la identificación de marcas de automóviles en imágenes, la publicidad dirigida y la monitorización de la presencia de marcas en redes sociales y medios digitales.

II. DESCRIPCION DEL DATASET

El conjunto de datos utilizado en este proyecto se compone de imágenes de logotipos de marcas de automóviles. Estas imágenes se recopilaron de un dataset de Kaggle (se encuentra en las referencias). El dataset contiene imágenes de 8 marcas de automóviles populares: Hyundai, Lexus, Mazda, Mercedes, Opel, Skoda, Toyota y Volkswagen.

El conjunto de datos se dividió en un conjunto de entrenamiento y un conjunto de validación. El preprocesamiento de imágenes incluyó la redimensión de todas las imágenes a 224x224 píxeles, normalización de píxeles para escalarlos en el rango [0, 1] y aumentos de datos como rotación, desplazamiento horizontal y vertical, y volteo horizontal.

Descripción de los archivos:

- ➔ Train: Es la carpeta para tomar las iamgenes de entrenamiento.
- ➔ Test: Es la carpeta para tomar las iamgenes de test/prueba.
- ➔ new_images : Es la carpeta para demostrar las predicciones.

III. ARQUITECTURA DEL MODELO DE DEEP LEARNING

El modelo utilizado en este proyecto se basa en la arquitectura VGG16, una red neuronal convolucional preentrenada en el conjunto de datos ImageNet. Al ser una red preentrenada, se beneficia de un aprendizaje transferido, lo que puede ser especialmente útil cuando se trabaja con un conjunto de datos específicos como el de logotipos de marcas de automóviles. Mientras que arquitecturas como ResNet y Inception pueden ser más eficientes y avanzadas, VGG16 es una elección “popular” para tareas de clasificación de imágenes debido a su balance entre profundidad, simplicidad y efectividad
Se eliminaron las capas superiores de VGG16 y se agregaron capas personalizadas para adaptar el modelo a la tarea de clasificación de marcas de automóviles. Las capas personalizadas incluyen una capa de aplanamiento (Flatten), una capa densa con 512 unidades y activación ReLU, una capa de dropout con una tasa de dropout del 50% para evitar el sobreajuste y una capa de salida con 8 unidades (una para cada marca de automóvil) y activación softmax.

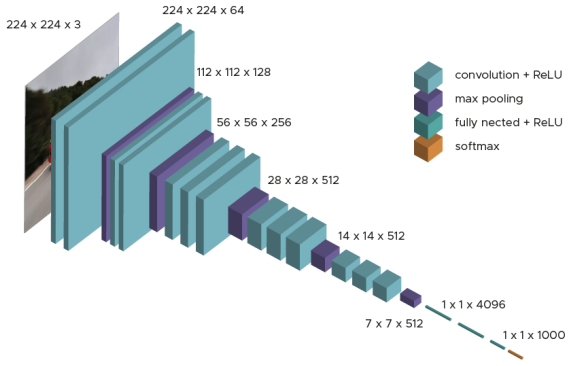

Figura 1.0. Estructura Base Algoritmo VGG16

```
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 224, 224, 3)]     0

block1_conv1 (Conv2D)        (None, 224, 224, 64)      1792

block1_conv2 (Conv2D)        (None, 224, 224, 64)      36928

block1_pool (MaxPooling2D)   (None, 112, 112, 64)      0

block2_conv1 (Conv2D)        (None, 112, 112, 128)     73856

block2_conv2 (Conv2D)        (None, 112, 112, 128)     147584

block2_pool (MaxPooling2D)   (None, 56, 56, 128)       0

block3_conv1 (Conv2D)        (None, 56, 56, 256)       295168

block3_conv2 (Conv2D)        (None, 56, 56, 256)       590080

block3_conv3 (Conv2D)        (None, 56, 56, 256)       590080

block3_pool (MaxPooling2D)   (None, 28, 28, 256)       0
...
Total params: 27564360 (105.15 MB)
Trainable params: 12849672 (49.02 MB)
Non-trainable params: 14714688 (56.13 MB)
```

Figura 2.0. Resumen del Modelo

IV. PROCESO DE ENTRENAMIENTO

El modelo se entrenó utilizando el optimizador Adam con una tasa de aprendizaje de 0.0001 y la función de pérdida de entropía cruzada categórica. El entrenamiento se realizó en dos etapas: la primera etapa incluyó el entrenamiento de las capas personalizadas agregadas al modelo, mientras que la segunda etapa implicó el ajuste fino (fine-tuning) de las últimas capas convolucionales de VGG16, con un learning rate más bajo (0.00001).

V. RESULTADOS Y ANALISIS

El modelo alcanzó una precisión de entrenamiento 79% y validación cercana al 77% en la primera etapa de entrenamiento, lo que indica un buen rendimiento inicial. Después de la etapa de ajuste fino, la precisión mejoró aún más, llegado a el 89% y 84% . La pérdida se redujo significativamente durante el entrenamiento, lo que indica que el modelo se está ajustando adecuadamente a los datos.

```
Epoch 1/10
78/78 [==============================] - 173s 2s/step - loss: 1.7839 - accuracy: 0.3728 - val_loss: 1.2054 - val_accuracy: 0.6484
Epoch 2/10
78/78 [==============================] - 176s 2s/step - loss: 1.2075 - accuracy: 0.6090 - val_loss: 0.9345 - val_accuracy: 0.7370
Epoch 3/10
78/78 [==============================] - 173s 2s/step - loss: 1.0752 - accuracy: 0.6497 - val_loss: 0.8418 - val_accuracy: 0.7604
Epoch 4/10
78/78 [==============================] - 173s 2s/step - loss: 0.9464 - accuracy: 0.6896 - val_loss: 0.8219 - val_accuracy: 0.7344
Epoch 5/10
78/78 [==============================] - 173s 2s/step - loss: 0.8730 - accuracy: 0.7263 - val_loss: 0.7596 - val_accuracy: 0.7734
Epoch 6/10
78/78 [==============================] - 173s 2s/step - loss: 0.8275 - accuracy: 0.7320 - val_loss: 0.7908 - val_accuracy: 0.7552
Epoch 7/10
78/78 [==============================] - 176s 2s/step - loss: 0.7832 - accuracy: 0.7437 - val_loss: 0.7482 - val_accuracy: 0.7734
Epoch 8/10
78/78 [==============================] - 173s 2s/step - loss: 0.7422 - accuracy: 0.7634 - val_loss: 0.7257 - val_accuracy: 0.7734
Epoch 9/10
78/78 [==============================] - 173s 2s/step - loss: 0.6877 - accuracy: 0.7759 - val_loss: 0.7402 - val_accuracy: 0.7734
Epoch 10/10
78/78 [==============================] - 183s 2s/step - loss: 0.6717 - accuracy: 0.7916 - val_loss: 0.6929 - val_accuracy: 0.7760
```

Figura 3.0. Resultados del Primer Modelo

```
Epoch 1/5
78/78 [==============================] - 268s 3s/step - loss: 0.5628 - accuracy: 0.8206 - val_loss: 0.6637 - val_accuracy: 0.8151
Epoch 2/5
78/78 [==============================] - 264s 3s/step - loss: 0.4327 - accuracy: 0.8634 - val_loss: 0.5890 - val_accuracy: 0.8229
Epoch 3/5
78/78 [==============================] - 256s 3s/step - loss: 0.3853 - accuracy: 0.8763 - val_loss: 0.5978 - val_accuracy: 0.8333
Epoch 4/5
78/78 [==============================] - 202s 3s/step - loss: 0.3401 - accuracy: 0.8936 - val_loss: 0.5913 - val_accuracy: 0.8385
Epoch 5/5
78/78 [==============================] - 210s 3s/step - loss: 0.3233 - accuracy: 0.8976 - val_loss: 0.5593 - val_accuracy: 0.8411
```

Figura 4.0. Resultados del Segundo Modelo

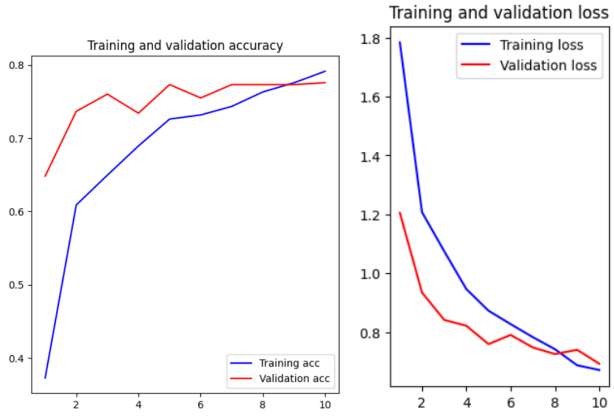

Figura 5.0. Grafica de Accuracy y Perdida para Train y Test (Primer Modelo)

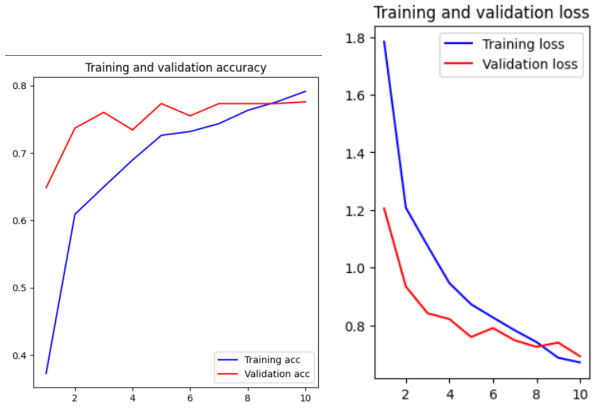

Figura 6.0. Grafica de Accuracy y Perdida para Train y Test (Segundo Modelo)

VI. MEJORAS Y AJUSTES REALIZADOS

Para mejorar el rendimiento del modelo, se aplicaron técnicas de ajuste fino en las últimas capas convolucionales de VGG16. Esto permitió al modelo adaptarse de manera más específica a las características de las imágenes de logotipos de automóviles. Además, se utilizó una capa de dropout para mitigar el sobreajuste.

VII. EJEMPLOS DE PREDICCIONES

Se realizaron pruebas con imágenes de logotipos de marcas de automóviles no vistas por el modelo durante el entrenamiento. El modelo pudo realizar predicciones precisas de la marca de automóvil en estas imágenes, lo que demuestra su capacidad para generalizar y reconocer marcas de automóviles desconocidas.

VIII. CONCLUSIONES

Se logró desarrollar un modelo de clasificación de marcas de automóviles con un alto rendimiento. Este modelo puede ser utilizado en aplicaciones del mundo real, como la identificación de marcas de automóviles en imágenes de tráfico, análisis de datos de redes sociales y más. Para futuros trabajos, se podría considerar la expansión del conjunto de datos con más marcas de automóviles y la exploración de arquitecturas de modelos más avanzadas para mejorar aún más el rendimiento.

IX. Referencias

[1] Dataset tomado de Kaggle: https://www.kaggle.com/datasets/volkandl/car-brand-logos

[2] Team, D. (2023, 30 octubre). *VGG: ¿Qué es este modelo? ¡Daniel te lo cuenta todo!* Formation Data Science | DataScientest.com. https://datascientest.com/es/vgg-que-es-este-modelo-daniel-te-lo-cuenta-todo

[3] Bressem, K. K., Adams, L. C., Erxleben, C., Hamm, B., Niehues, S. M., & Vahldiek, J. L. (2020). Comparing different deep learning architectures for classification of chest radiographs. Scientific Reports, 10(1). https://doi.org/10.1038/s41598-020-70479-z