

# Análisis y Reporte sobre el desempeño del modelo

Diego Emilio Barrera Hernández

A01366802

## Introducción:

Se realizó un modelo de clasificación categórica para identificar 3 categorías, y con estos clasificarlos según su imagen y su nombre. Las clases son las siguientes:

- Humano
- Anime
- Cartoon

Para mejor visualización de las imágenes, se puede consultar directamente las páginas de referencia dónde se obtuvieron dichas imágenes, en la siguiente URL:

- Humanos

<https://www.kaggle.com/datasets/ahmadahmadzada/images2000>

- Anime

<https://www.kaggle.com/datasets/diraizel/anime-images-dataset>

- Cartoon

<https://www.kaggle.com/datasets/hadipratamatulili/anime-vs-cartoon-vs-human>

## Método empleado:

Para este análisis, se empleó un modelo de red neuronal basado en la arquitectura VGG16 para la clasificación. Se implementó un conjunto de aumento de datos durante el entrenamiento y la validación, utilizando la

librería de ImageDataGenerator. La red neuronal donde se añaden capas de aplanamiento, seguidas de una capa densa con 256 unidades y activación "relu", una capa de normalización por lotes, y una capa de abandono con una tasa del 50%. Finalmente, se incluye una capa de salida con activación "softmax" para clasificar en las 3 categorías: Anime, Humano y Cartoon.

A continuación, explico la razón de las configuraciones anteriores. La capa Flatten se eligió para preparar los datos tridimensionales para la entrada en las capas densas. La capa con activación relu de 256 neuronas permite aprender combinaciones complejas de las características de nuestros datos. La capa de normalización estabiliza el entrenamiento, mientras que la capa de dropout se usa para evitar el sobreajuste. Finalmente, la capa softmax se utiliza para clasificar nuestras 3 categorías.

Esta información se respalda en el artículo "Transfer Learning for Classification of Fruit Ripeness Using VGG16", que explica la importancia de la distribución de las capas en un modelo de clasificación de imágenes. El artículo muestra gráficamente ejemplos de iniciar con una capa Flatten y terminar con una capa Softmax, la cual se encarga de la tarea de clasificación. Además, se aborda la capa Dropout y su importancia para evitar el overfitting. Según el autor, "Dropout es una técnica de regularización que puede ayudar a las CNN a resistir el overfitting y acelerar el proceso de entrenamiento. Funciona eliminando temporalmente capas ocultas y visibles distribuidas aleatoriamente en la red, y redirigiendo a las neuronas bien entrenadas para reducir el aprendizaje interdependiente en cada neurona" [1]

## Métricas utilizadas:

En un estudio sobre CNNs, la métrica "accuracy" es útil para evaluar su rendimiento en la clasificación de imágenes entre humanos, anime y cartoon. Mide la proporción de imágenes correctamente clasificadas, siendo simple y efectiva para esta tarea. Es importante considerar que la accuracy no siempre es la mejor métrica para todas las aplicaciones de clasificación de imágenes. [2]

En la primera prueba del modelo, se empleó un total de 15 épocas. Durante este proceso, se observaron mejoras progresivas en la precisión tanto en los datos de entrenamiento como en los de validación. En la primera época, el modelo comenzó con un accuracy del **49.00%** en el entrenamiento y del **71.40%** en la validación. Con el transcurso de las épocas, se evidenció un crecimiento significativo en la capacidad predictiva del modelo, alcanzando una precisión del **82.80%** en el entrenamiento y del **83.60%** en la validación en la última época.

Los valores de pérdida también disminuyeron de manera consistente a lo largo de las épocas, lo que indica una mejora en la capacidad del modelo para predecir con mayor precisión. Al inicio, la pérdida fue de **1.3575** en el entrenamiento y de **0.7569** en la validación, mientras que al final del proceso, estos valores disminuyeron significativamente a **0.4327** en el entrenamiento y **0.4260** en la validación.

Se hicieron 3 representaciones gráficas de nuestros resultados. En la Fig 1, se pueden

ver los resultados obtenidos en el entrenamiento y la validación.

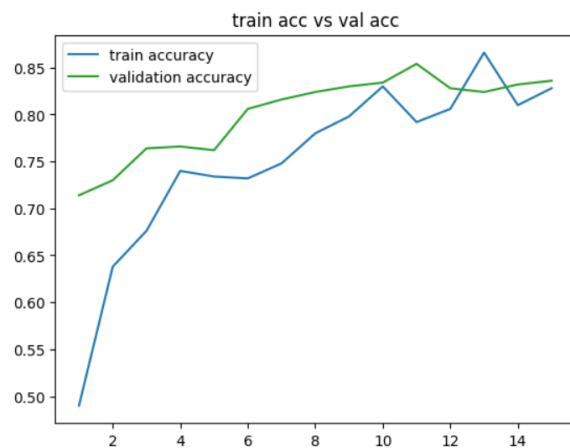


Fig 1: Evolución de precisión de entrenamiento y validación

En esta gráfica podemos observar cómo evolucionan la precisión de entrenamiento y la de validación. Se puede ver que la precisión de entrenamiento comienza en 0.72 y aumenta hasta 0.83 a medida que el modelo se entrena. La precisión de validación, por su parte, comienza en 0.49 y también aumenta hasta 0.82. Este es un buen resultado, pero es importante señalar que la precisión de entrenamiento es ligeramente superior a la de validación, lo cual podría indicar que el modelo está sobreajustándose a los datos de entrenamiento. Sin embargo, podemos observar que la precisión de validación superó a la de entrenamiento entre las épocas 12 y 14, lo que sugiere que el modelo se estaba regulando de manera efectiva. Es importante mencionar que esta diferencia en los valores no solo se puede deber al sobreajuste, sino también a la complejidad de los datos de entrenamiento, lo cual tiene sentido dada la distribución del dataset.

Ahora presentamos la Fig 2, aquí se puede observar los resultados visuales de la pérdida en entrenamiento y la pérdida en validación.

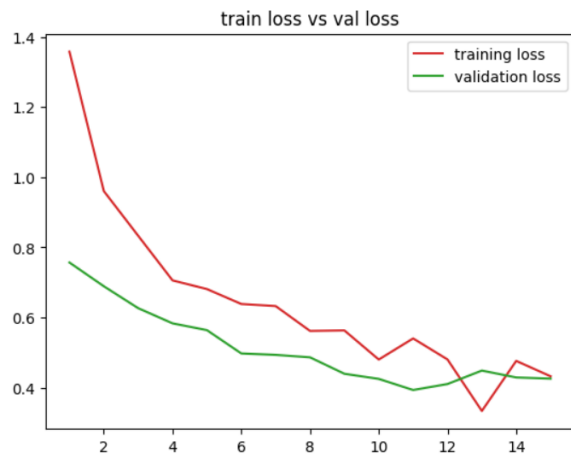


Fig 2: Evolución de la pérdida de entrenamiento y validación

La pérdida de entrenamiento comienza en 1,4 y disminuye hasta 0,43 a medida que el modelo se entrena. La pérdida de validación comienza en 0,75 y también disminuye hasta aproximadamente 0,43. Como se mencionó previamente, La pérdida de entrenamiento y validación disminuye a medida que el modelo se entrena, lo que indica que el modelo está aprendiendo a realizar las tareas para las que fue diseñado.

Por último, en la Fig 3, podemos observar una matriz de confusión que nos da los resultados de verdaderos positivos, verdaderos negativos, falsos positivos y falsos negativos.

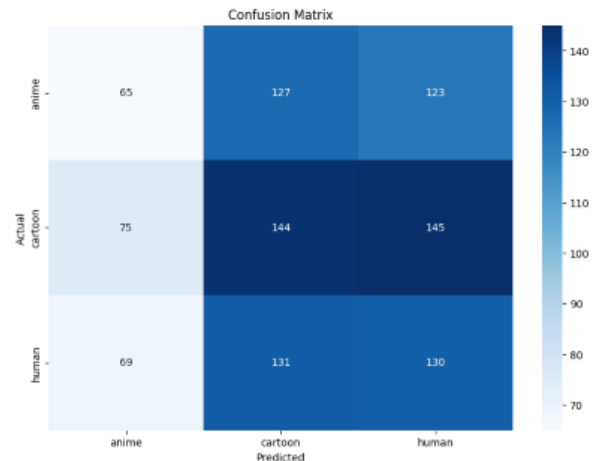


Fig 3: Matriz de confusión de los datos de train

En la matriz podemos ver los resultados de nuestras 1009 imágenes de prueba. En este caso, Anime ha sido la categoría más afectada, ya que solo se predijeron 65 imágenes correctamente. Por otro lado, tenemos 144 predicciones correctas para Cartoon y 130 para Human. El resto fueron predicciones erróneas. Cabe mencionar que Cartoon fue confundida con Human 145 veces, siendo esta la mayor confusión. Esto puede deberse a la distribución del dataset.

En esta última figura podemos ver en termino de porcentajes, los resultados de las predicciones.

	precision	recall	f1-score	support
anime	0.82	0.72	0.77	315
cartoon	0.82	0.94	0.88	364
human	0.85	0.81	0.83	330
accuracy			0.83	1009
macro avg	0.83	0.82	0.82	1009
weighted avg	0.83	0.83	0.83	1009

Fig 4: Tabla de resultados de test

La matriz muestra una precisión general del 83%, con precisiones por clase de 82% para 'anime', 82% para 'cartoon' y 85% para 'human'. El recall es especialmente bajo para

'anime' (77%), indicando problemas en identificar correctamente esta categoría. Los F1-scores bajos reflejan un equilibrio pobre entre precisión y recall. Esto podría significar que el modelo necesita mejoras en su configuración y datos de entrenamiento para mejorar su rendimiento.

## Mejora del modelo:

Al igual que en el primer modelo, se utilizó una red neuronal basada en la arquitectura VGG16. Solo se hicieron algunos cambios en la arquitectura del modelo. En esta ocasión, se usaron dos capas densas: una de 512 neuronas con activación ReLU y otra de 256 neuronas con activación ReLU. Es importante mencionar que se eliminaron las capas Dropout, ya que estaban impidiendo un aprendizaje correcto y se demostró que no eran del todo necesarias, dado que no se identificó un overfitting considerable en la configuración del modelo anterior. Además, se cambiaron las características del ajuste del modelo, incrementando a 25 los steps por cada época, con un total de 30 épocas; finalmente, para la validación, se utilizaron 25 steps.

En este caso, en la primera época, el modelo comenzó con un accuracy del **71.00%** en el entrenamiento y del **64.83%** en la validación. Con el transcurso de las épocas, se evidenció un crecimiento significativo en la capacidad predictiva del modelo, alcanzando una precisión del **86.25%** en el entrenamiento y del **84.17%** en la validación en la última época.

Los valores de pérdida también disminuyeron de manera consistente a lo

largo de las épocas, lo que indica una mejora en la capacidad del modelo para predecir con mayor precisión. Al inicio, la pérdida fue de **0.8354** en el entrenamiento y de **1.0023** en la validación, mientras que al final del proceso, estos valores disminuyeron significativamente a **0.3387** en el entrenamiento y **0.4250** en la validación.

En la Fig. 5 podemos observar los resultados de nuestra matriz de confusión.

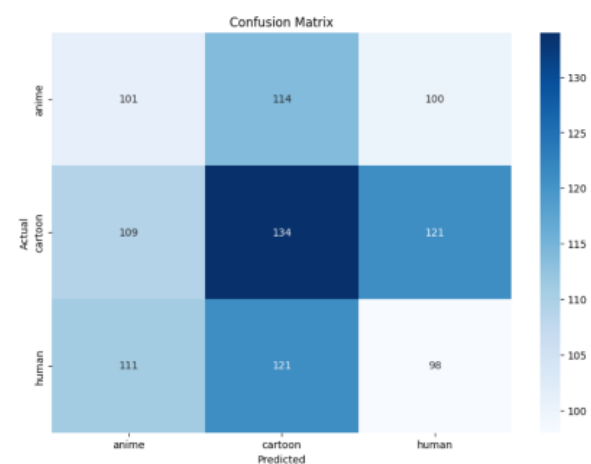


Fig 5: Matriz de confusión de los datos de train de la mejora de modelo

En la matriz podemos ver los resultados de nuestras 1009 imágenes de prueba. En este segundo caso, "Human" ha sido la categoría más afectada, ya que solo se predijeron 98 imágenes correctamente. Por otro lado, tenemos 134 predicciones correctas para "Cartoon", lo que son 10 predicciones correctas menos que en el modelo anterior, y 101 para "Anime", lo que son 36 predicciones correctas más que en el modelo pasado. El resto fueron predicciones erróneas. Cabe mencionar que "Cartoon" y "Human" tuvieron bastantes confusiones entre ellas, siendo estas las mayores confusiones. Esto puede deberse a la

distribución del dataset o a la falta de diferencias en las imágenes.

Por último, en la Fig. 6 se observa los resultados finales de las predicciones.

	precision	recall	f1-score	support
anime	0.77	0.82	0.79	315
cartoon	0.90	0.88	0.89	364
human	0.88	0.84	0.86	330
accuracy			0.85	1009
macro avg	0.85	0.85	0.85	1009
weighted avg	0.85	0.85	0.85	1009

Fig 6: Tabla de resultados de test de la mejora del modelo

La matriz muestra una precisión general del 85%, con precisiones por clase de 77% para 'anime', 90% para 'cartoon' y 88% para 'human'. El recall es bajo para 'anime' (79%), indicando problemas en identificar correctamente esta categoría. Estos resultados nos pueden decir dos cosas importantes, la primera es la falta de balanceo de los datos y la segunda es la oportunidad de una mejora continua del modelo, pues significa que aún se puede ajustar para poder tener predicciones aún más correctas. Finalmente, con estos resultados podemos observar que el segundo modelo mejoró a grandes rasgos, sin mencionar la precisión de Anime, pues esa empeoró de manera no significativa.

## Conclusión

Comparando ambos modelos, podemos ver una mejora de **4.55%** en accuracy y una disminución de error del -0.0940, por lo que contamos con un mejor modelo, el cual, nos llega a dar mejores predicciones.

Cabe destacar que una de las razones de las confusiones entre "Anime" y "Cartoon" se debe al parentesco entre ambas clases. El modelo podría estar buscando características diferentes para poder aprender de ambas clases; una de ellas puede ser la distribución o posición de los personajes.

## Resultados

[https://drive.google.com/drive/folders/1C53YiMKrdIL0Vwqmf2fs\\_HEHTxAAYlBz?usp=sharing](https://drive.google.com/drive/folders/1C53YiMKrdIL0Vwqmf2fs_HEHTxAAYlBz?usp=sharing)

Modelo	Acc Train	Acc val	Acc Test
M1	82.80	83.60	83%
M2	86.25	84.17	85%

## Referencias

- [1] A. N. Hermana, D. Rosmala, and M. G. Husada, "Transfer Learning for Classification of Fruit Ripeness Using VGG16," in ICCMB, 2021, pp. 8.
- [2] T. Deng, "A Survey of Convolutional Neural Networks for Image Classification: Models and Datasets," in 2022 International Conference on Big Data, Information and Computer Network (BDICN), 2022, pp. 11.