

Aplicación de Redes Neuronales Convolucionales para la Clasificación de Imágenes

Dora Alicia Guevara Villalpando
Matrícula: 1551003

Universidad Autónoma de Nuevo León)
Facultad de Ciencias Físico Matemáticas
Maestría en Ciencia de Datos
Procesamiento y Clasificación de Datos

dora.guevaravll@uanl.edu.mx

Resumen—Este trabajo presenta una aplicación de redes neuronales convolucionales (CNN) para la clasificación de imágenes en tres categorías: aves, gatos y perros. Se empleó el preprocesamiento de datos mediante normalización y aumento de datos para mejorar la capacidad de generalización del modelo. La red neuronal fue entrenada con el optimizador Adam y la función de pérdida *categorical crossentropy*. Los resultados obtenidos muestran una precisión de validación del 86 % en las mejores iteraciones, lo que indica un buen desempeño del modelo en la tarea de clasificación.

I. INTRODUCCIÓN

Las redes neuronales convolucionales han demostrado ser altamente eficaces en tareas de visión por computadora, como la clasificación de imágenes, la detección de objetos y la segmentación. En este proyecto, se desarrolla un modelo CNN para la clasificación de imágenes de animales en tres clases distintas. La aplicación de técnicas de preprocesamiento y aumento de datos permite mejorar la robustez del modelo, evitando el sobreajuste y aumentando su capacidad de generalización.

El objetivo de este trabajo es evaluar el desempeño de un modelo CNN en la clasificación de imágenes de animales y analizar el impacto del preprocesamiento y el aumento de datos en su precisión y capacidad de generalización.

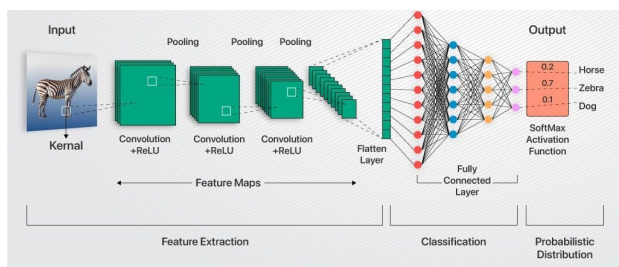


Figura 1. Ejemplo de una red neuronal convolucional.

II. METODOLOGÍA

II-A. Preprocesamiento de datos

- **Organización del conjunto de datos:** Se recopilaron imágenes de las categorías aves, gatos y perros, las cuales fueron organizadas en carpetas para facilitar su uso en entrenamiento y prueba.
- **División de datos:** Se estableció una proporción del 80 % de las imágenes para entrenamiento y 20 % para prueba.
- **Normalización de imágenes:** Los valores de los píxeles fueron escalados dividiéndolos por 255, lo que ayuda a mejorar la eficiencia del entrenamiento al reducir la variabilidad numérica.
- **Verificación de imágenes:** Se implementó un proceso para identificar imágenes corruptas o con formatos incompatibles, asegurando que solo se usaran imágenes válidas.

II-B. Aumento de datos

- **Transformaciones aplicadas:** Se utilizaron técnicas como rotaciones aleatorias, desplazamientos horizontales y verticales, cambios de escala y volteos horizontales para incrementar la variabilidad en los datos.
- **Propósito del aumento de datos:** Ayuda a mejorar la capacidad de generalización del modelo y a reducir el sobreajuste.

En la figura 2 se muestra un ejemplo de como queda la imagen después del aumento.



Figura 2. Ejemplo de imagen original y transformada con aumento de datos.

II-C. Arquitectura del modelo

- **Capas convolucionales:** Se utilizaron tres capas convolucionales con activación *ReLU*, cada una seguida de una capa de *max-pooling* para reducir la dimensionalidad.
- **Capas densas:** Se agregaron capas completamente conectadas con activación *ReLU* y una capa de salida con activación *softmax* para la clasificación en tres categorías.
- **Regularización:** Se implementó una capa de *Dropout* para reducir el sobreajuste.

II-D. Entrenamiento y validación

- **Hiperparámetros:** Se configuró el entrenamiento con 100 épocas, un tamaño de lote de 16 y el optimizador *Adam*.
- **Función de pérdida:** Se utilizó *categorical crossentropy* para medir la diferencia entre las predicciones y las etiquetas reales.
- **Monitoreo del entrenamiento:** Se registraron la precisión y la pérdida en cada época para evaluar la evolución del modelo.

III. RESULTADOS

- La precisión en el conjunto de entrenamiento aumentó progresivamente, alcanzando valores superiores al 80 % en algunas épocas.
- La precisión de validación alcanzó un valor máximo de 86 %.
- La función de pérdida disminuyó de manera constante tanto en entrenamiento como en validación.
- Se generaron gráficas de evolución de precisión y pérdida.

En la figura 3 se muestra la evolución de la precisión y la pérdida durante el entrenamiento.

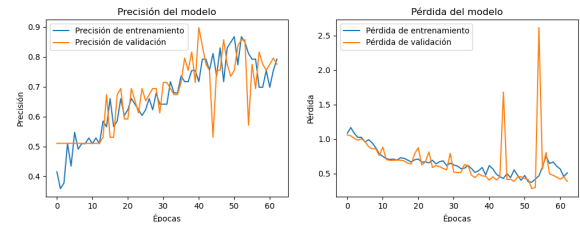


Figura 3. Curvas de precisión y pérdida durante el entrenamiento.

IV. CONCLUSIONES

El modelo CNN desarrollado demostró ser efectivo en la clasificación de imágenes de animales, alcanzando una alta precisión en la validación. El uso de técnicas de preprocesamiento y aumento de datos contribuyó a mejorar la capacidad de generalización del modelo. Para futuras mejoras, se podría aumentar el volumen de datos de entrenamiento o utilizar arquitecturas preentrenadas para optimizar el rendimiento del modelo. En general, los resultados obtenidos validan la eficacia de las CNN en problemas de clasificación de imágenes.

En la figura 4 se muestra la clasificación de una imagen que no formó parte de la base inicial de imágenes.

Predicción: <function predict_image at 0x169c7fd80>



Figura 4. Ejemplo de predicción del modelo en imágenes nuevas.