Universidad Francisco Marroquín

Machine Learning Models

José Alejandro Guzmán Zamora

Art Classification – Redes Neuronales Convolucionales

Introducción

Las arquitecturas de redes neuronales son útiles especialmente en el caso de análisis de imágenes, esto es porque al momento de utilizar un conjunto de imágenes como dataset, cada píxel es equivalente a lo que se considera una feature. Esto indica que la cantidad de variables que ingresan a un modelo pueden crecer rápidamente y modelos más sencillos de Machine Learning tales como la regresión lineal empiezan a tener problemas al intentar hacer predicciones correctas. El problema con las redes neuronales para la clasificación de imágenes es que no hay relación espacial entre los píxeles que ingresan a la red. Esto puede aumentar la cantidad de pesos a cambiar en cada iteración e incluso puede causar que la red se los memorice a falta de encontrar relaciones directas entre los mismos. Para solucionar estos detalles, en este caso se utilizan las redes neuronales convolucionales (CNN). Las arquitecturas de las CNNs contienen capas de un número arbitrario de filtros que se pasan en cada imagen. Esto produce feature maps, imágenes que capturan una relación espacial entre los píxeles y pueden llegar a detectar detalles específicos en imágenes. En este caso lo que se desea realizar es clasificar un conjunto de imágenes de arte. Se utilizó el dataset "Art Images" del sitio Kaggle, este contiene un total de 8,685 imágenes. Las imágenes están divididas en las siguientes categorías: *Drawing, Painting, Sculptures, Ionography* y *Engravings.* Antes de empezar a hacer el análisis correspondiente y la construcción del modelo se tuvo que adaptar el dataset a arreglos de numpy para su ingreso a los modelos posteriores. Para hacer esta adaptación se recorrió la estructura de directorios original, insertando cada una de las imágenes en un arreglo y su label correspondiente en otro arreglo. Vale la pena mencionar que se tuvieron que unir las clases de dibujo y gravados para balancear el dataset, esta decisión se tomó considerando que ambas clases son similares y la suma de imágenes de ambas es equivalente a lo que las demás clases poseen. Posteriormente se utilizó la función resize() de OpenCV para establecer un tamaño uniforme de las imágenes, se configuró un tamaño de 64 por 64 píxeles en 3 canales. Este tamaño fue el ideal para el entrenamiento del modelo porque incluye el detalle suficiente para que haya una diferenciación clara entre las clases, pero no al punto que el rendimiento de entrenamiento disminuye y el modelo empieza a caer en alta varianza.

Arquitectura de la red y entrenamiento

Como todo proceso de Machine Learning, hubo una gran cantidad de pruebas con diferentes arquitecturas y cambiando ciertos parámetros en base a los resultados y a la historia de entrenamiento de cada modelo. Por ejemplo, en una prueba se utilizó un total de 2 capas convolucionales y la misma cantidad de capas densas para realizar la predicción. Esta distribución de capas en conjunto con valores relativamente bajos de dropout causaron que la diferencia entre el Train Accuracy y el Validation Accuracy aumentara desde iteraciones tempranas, esto acompañado de una exactitud por debajo de 80% que claramente se deseaba mejorar. Después se intentó hacer lo opuesto, se redujeron la cantidad de neuronas por cada capa densa y se aumentó la cantidad de capas convolucionales a cinco. Esta distribución demostró reducir ligeramente el problema de la varianza, pues tanto el Train como el Validation Accuracy crecieron de manera similar, sin embargo, la historia del entrenamiento mostró altos grados de diferencia entre las exactitudes de cada iteración. Al igual que el modelo anterior, este tuvo una exactitud por debajo del 80% y se decidió continuar haciendo cambios para resolver los problemas que se presentaron. La última arquitectura antes de la que se utilizó tuvo un mejor rendimiento que las anteriores, esto se debe a que se utilizaron 5 capas convolucionales seguidas de 2 capas densas con un número de neuronas entre los utilizados para los modelos anteriores. El entrenamiento de este modelo es visualmente superior a los anteriores y el crecimiento de la curva de exactitud para Train y Validation fue considerablemente similar, es decir hubo poco

overfitting para la cantidad de épocas (25). La exactitud de prueba a la que se logró llegar con este modelo fue de 84%, este modelo logró ascender sobre la planicie en la que los modelos anteriores se habían quedado estancados. El problema de esta arquitectura es que no sigue las convenciones de uso de las CNN, esto es porque la cantidad de filtros es mayor al principio y va disminuyendo conforme pasan las capas. Si bien es cierto hacer esto no es algo estrictamente incorrecto, esta estructura disminuye la cantidad de parámetros que se están tomando en cuenta en capas profundas de la red, especialmente en las últimas capas densas. De igual manera, es lógico que lo más conveniente es comenzar con un número arbitrario de filtros y aumentarlo conforme pasan las capas porque cada vez que se hace una reducción de dimensionalidad con las capas de MaxPooling el aumento en la cantidad de filtros se compensa por la menor cantidad de operaciones que se deben de realizar. Además, en capas profundas hay mayor cantidad de filtros que con dimensionalidad baja pueden detectar detalles específicos de cada imagen.

Tomando en cuenta lo mencionado anteriormente se decidió utilizar una arquitectura con 5 capas convolucionales que comienzan con 32 filtros y van aumentando hasta 128 filtros en la última. En la **Figura 1** se puede verificar cada una de las capas utilizadas, hay una capa de MaxPooling por cada 2 capas convolucionales aproximadamente. Además, se agregó una capa de dropout con un valor de 25% para cada paso posterior a una reducción de dimensionalidad y también se incluyó una entre las capas densas con un valor de 30% para reducir la posibilidad de overfitting en esas últimas capas. De acuerdo con el modelo, al final hay un total de 1.5 millones de parámetros "entrenables", un millón más que el del modelo anterior. La diferencia es que la mayor cantidad de entrenamiento de estos parámetros es sobre imágenes de dimensiones bajas por lo que el rendimiento al momento de entrenar es similar. Se entrenó el modelo a lo largo de 30 epochs con el total de imágenes en cada una de estas iteraciones, pero separado en grupos de 64 imágenes. La gráfica que presenta la historia de entrenamiento se puede visualizar en la **Figura 2,** la separación entre la exactitud de entrenamiento y de validación es menor comparada a modelos anteriores y el crecimiento es uniforme. Sin embargo, después de la iteración 25 esta separación empieza a aumentar por lo que se decidió parar el entrenamiento en 30. La exactitud que este modelo obtuvo es de 87% sobre los datos de prueba, una cantidad considerablemente mayor a las obtenidas anteriormente y aceptable en el contexto de la clasificación. En la **Figura 3** se puede visualizar la matriz de confusión pertinente al modelo entrenado, en términos generales se puede ver un rendimiento adecuado, no obstante, sí se nota que la mayor cantidad de error corresponde a la clase de dibujo. Es decir, hay ciertas imágenes de otras clases que están siendo clasificadas como dibujo, especialmente la de esculturas. En el análisis de los feature maps se indaga más sobre la posible razón de este error.

Feature Maps y Análisis

Una vez entrenado y evaluado el modelo, se inició con el análisis de feature maps, como se mencionó al inicio, estas imágenes son el resultado de multiplicar cada uno de los filtros que hay en cada capa convolucional. Esto quiere decir que al ingresar una imagen en el modelo cada una de las capas convolucionales va a devolver un total de feature maps correspondientes a la cantidad de filtros que se configuraron y la dimensionalidad depende de si se agrega algún tipo de padding alrededor de la imagen. En este caso no se agregó padding, por lo que cada capa convolucional subsecuente tiene una dimensionalidad de (n-2, n-2) si (n, n) es la dimensionalidad de la imagen original. Es importante notar que con estas capas subsecuentes se refieren a los casos en los que hay una capa convolucional seguida de otra directamente, los demás casos incluyen capas de MaxPooling que reducen en mayor grado estos valores. Con el objetivo de encontrar las diferencias en la clasificación de los distintos tipos de imágenes, el estudio de feature maps que se por clase. Se obtuvo una imagen del internet que se considera pertenece a cada una de las clases, se introdujo en el modelo, se verificó que se clasificara adecuadamente y después se empezaron a analizar los feature maps internos a la red. Tal como se mencionó en la sección de la arquitectura, el modelo tiene 5 capas convolucionales, sin embargo, por la cantidad de feature maps que cada una de estas producen, se decidió hacer comparar solamente la capa 1, 3 y 4. Aunque no se tenga un panorama totalmente completo de los feature maps, esta selección de capas es la correcta pues en la primera podemos visualizar el comienzo con 32 feature maps, en la tercer capa se puede hacer el mismo análisis pero con 64 feature maps y en la cuarta con 128.

Se inició el análisis con la clase de pintura, la imagen que se utilizó de ejemplo de la imagen de la Mona Lisa y el modelo la clasificó exitosamente como *painting.* Los feature maps de la primera capa convolucional muestran un recorrido relativamente simple de la imagen. En el lado izquierdo de la **Figura 4** muestra los cuatro feature maps de 32 que se consideraron importantes, en las primeras dos imágenes se puede ver claramente que hay una detección de líneas verticales y diagonales. Al ser esta la primera capa, la detección de características es general y cada uno de los feature maps todavía tiene una dimensionalidad relativamente grande que le permite a los posteriores detectar detalles específicos. Las siguientes 2 imágenes de este lado izquierdo muestran lo que parece ser un poco de contraste entre la figura humana principal y el fondo, además de un poco de profundidad en el área de los ojos. En la parte derecha de la **Figura 4** se presentan los resultados de la segunda convolución, de igual manera son cuatro feature maps importantes de los 64 de esta convolución. En estas imágenes se puede ver que hay un enfoque más fuerte en la piel de la figura humana central y en todas hay una detección directa de los ojos. En base a estos feature maps se podría decir que al ser las pinturas generalmente representaciones humanas, la detección se basa en identificar detalles tales como los ojos, contornos y el contraste de la piel en la imagen.

En la **Figura 5** se muestran los resultados de las primeras dos convoluciones para una imagen que pertenece a la clase de dibujo. Al igual que en la clase anterior, la primera convolución es una detección general de características tales como líneas. La diferencia con esta clase es que la naturaleza de la imagen causa que la detección sea en el centro de esta. Este fenómeno ocurre porque la mayoría de las imágenes de esta clase son de dibujos a papel y lápiz y por lo general estos dibujos tienen su actividad en el centro de una página y el resto es blanco. Este detalle se puede notar con más claridad en la imagen superior derecha de la **Figura 5** correspondiente a la segunda convolución, se puede notar que se resalta la parte blanca de la imagen, es decir, todo aquello que no tiene trazos de lápiz o del instrumento que se utilice. En el caso del tipo de iconografía el análisis es relativamente sencillo, esto es porque la mayoría de estas imágenes comparten una decoración al estilo barroco alrededor de una figura humana central y que tiene un tipo de figura circular alrededor de la cabeza. En la **Figura 6** se puede ver que la mayoría de los feature maps se enfocan en la figura circular mencionada y en los detalles del marco de la imagen. Por último, se aplican las convoluciones para el tipo de esculturas. La imagen de ejemplo que se utiliza es la de la escultura del gigante de Cayalá en la Ciudad de Guatemala. En las cuatro imágenes del lado izquierdo de la **Figura 7** se puede verificar que hay detección de líneas verticales y horizontales, esta detección es evidente dado que el fondo de la imagen presenta un edificio con ventanas. Lo que es interesante de esta primera convolución es que se puede decir que la atención de la detección está en el fondo de la imagen, no necesariamente en la escultura directamente. Este detalle es más claro en las siguientes cuatro imágenes que se presentan, hay un enfoque especial en el área de la grama de la imagen. De igual manera, las convoluciones más profundas se dedican a establecer relaciones detalladas del edificio del fondo y del contraste de la grama con el de la escultura.

Conclusiones y Recomendaciones

A lo largo del proceso de transformaciones necesarias, construcción de la arquitectura y entrenamiento del modelo de redes neuronales convolucionales se determinó que un número adecuado de convoluciones para la tarea de clasificación de imágenes de arte es cinco, esto es siempre y cuando la cantidad de filtros se configure de manera ascendiente. Asimismo, es fundamental utilizar capas Dropout para reducir la varianza y evitar que las redes memoricen características presentes solo en los datos de entrenamiento. En cuanto al análisis de feature maps, se hizo una visualización por cada clase para verificar los detalles que predominan en cada una. La idea de las redes convolucionales es simular la manera en la que los humanos identifican objetos por medio de la vista y la visualización de los features maps verdaderamente muestran la lógica detrás del algoritmo. Las pinturas se manifiestan en características relacionadas a las figuras humanas presentes en la mayoría de las imágenes, los dibujos en un centro relleno y un borde blanco que representa la hoja en la que el mismo se dibujó, las obras iconográficas se manifiestan en los detalles decorativos que rodean la imagen y las esculturas tienen un enfoque en el entorno y el fondo de la imagen. Se recomienda aumentar el número de imágenes con el objetivo de poder generalizar más y aumentar la exactitud del modelo. Además, se podría estructurar el dataset o crear uno nuevo con el objetivo de hacer un análisis y comparación por período histórico.

Anexos

Figura 1. Modelo Utilizado

```
Model: "sequential_5"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_21 (Conv2D)           (None, 62, 62, 32)        896
_________________________________________________________________
max_pooling2d_13 (MaxPooling (None, 31, 31, 32)        0
_________________________________________________________________
dropout_13 (Dropout)         (None, 31, 31, 32)        0
_________________________________________________________________
conv2d_22 (Conv2D)           (None, 29, 29, 64)        18496
_________________________________________________________________
conv2d_23 (Conv2D)           (None, 27, 27, 64)        36928
_________________________________________________________________
max_pooling2d_14 (MaxPooling (None, 13, 13, 64)        0
_________________________________________________________________
dropout_14 (Dropout)         (None, 13, 13, 64)        0
_________________________________________________________________
conv2d_24 (Conv2D)           (None, 11, 11, 128)       73856
_________________________________________________________________
conv2d_25 (Conv2D)           (None, 9, 9, 128)         147584
_________________________________________________________________
max_pooling2d_15 (MaxPooling (None, 4, 4, 128)         0
_________________________________________________________________
dropout_15 (Dropout)         (None, 4, 4, 128)         0
_________________________________________________________________
flatten_5 (Flatten)          (None, 2048)              0
_________________________________________________________________
dense_13 (Dense)             (None, 512)               1049088
_________________________________________________________________
dropout_16 (Dropout)         (None, 512)               0
_________________________________________________________________
dense_14 (Dense)             (None, 256)               131328
_________________________________________________________________
dense_15 (Dense)             (None, 4)                 1028
=================================================================
Total params: 1,459,204
Trainable params: 1,459,204
Non-trainable params: 0
```

Figura 2. Historia de Entrenamiento del Modelo Utilizado

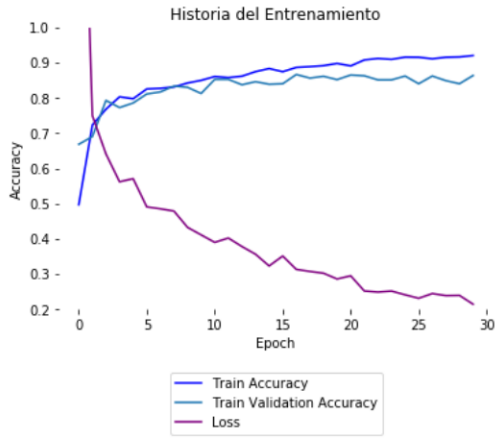
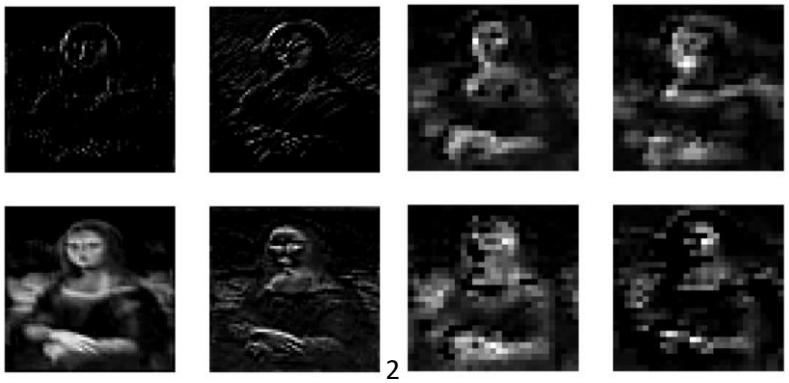


Figura 3. Matriz de Confusión

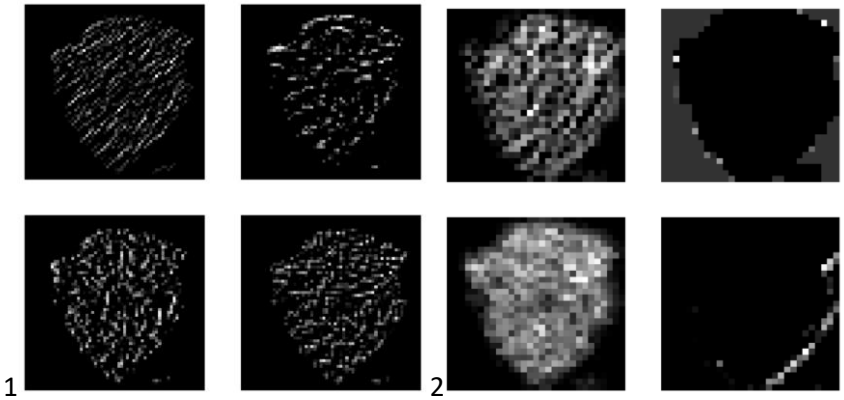
	Drawings	Iconography	Painting	Sculpture
Drawings	19%	1.6%	1.5%	2.3%
Iconography	0.7%	25%	0.8%	0.3%
Painting	0.8%	1%	23%	1%
Sculpture	1.8%	0.6%	0.4%	20%

Figura 4. Painting Convoluciones 1 y 2



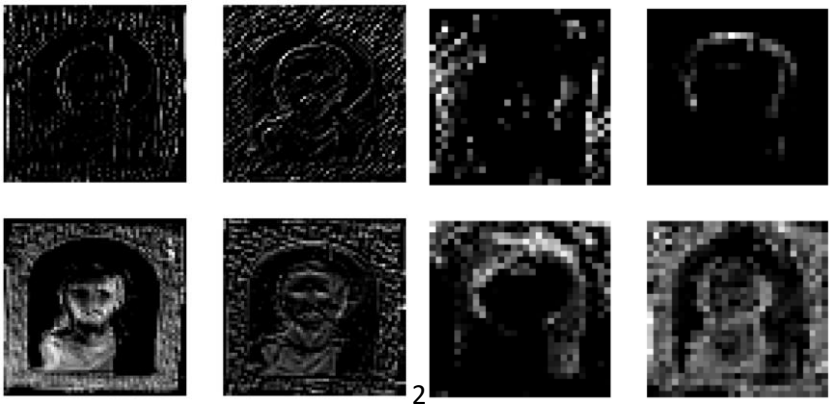
1 2

Figura 5. Drawing Convoluciones 1 y 2



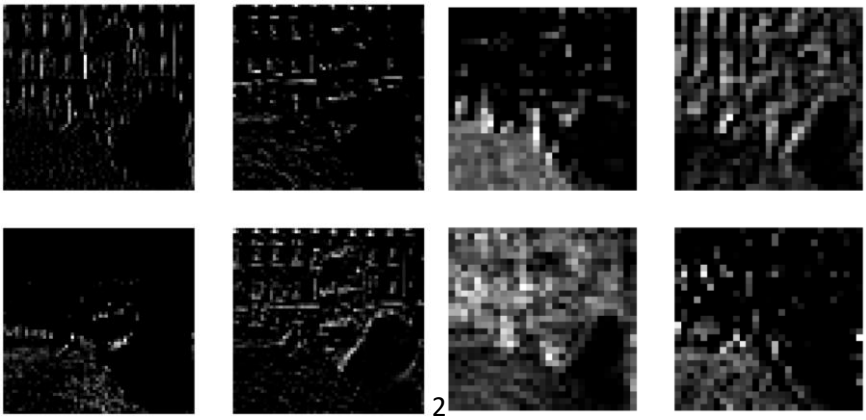
1 2

Figura 6. Iconography Convoluciones 1 y 2



1 2

Figura 7. Sculpture Convoluciones 1 y 2



1 2