

Proyecto Final Algoritmos 2022-2



Grupo: 1

Integrantes:

- Bryan Smith Colorado Lopez
- Miguel Angel Puentes Cespedes
- Edgar Daniel Gonzalez Diaz
- Jhonatan Steven Rodriguez Ibanez
- Obed Felipe Espinosa Angarita

Problema:

Investigar métodos de clasificación de imágenes mediante redes neuronales y la implementación de grafos representados por medio de listas de adyacencia, principalmente enfocado hacia convolución y algunas capas superiores.

Objetivos:

- Identificar un método que permita la transcripción de imágenes a grafos útiles como entrada (input) a una red neuronal de clasificación
- Diseñar la estructura de un prototipo de red neuronal que permita clasificar un conjunto de imágenes mediante Deep Learning
- Implementar un modelo de convolución y simplificación de imágenes mediante el uso de grafos

Origen de los datos:

Los datos se pueden extraer de datasets de imágenes (principalmente enfocado tentativamente hacia la identificación de animales en una imagen, o bien dígitos en la misma), por otra parte, dada la necesidad de reducir el problema a imágenes pequeñas, algunas imágenes podrán ser diseñadas por el equipo de trabajo de ser necesario (tratándose de imágenes de largos y anchos no mayores a 10 unidades o píxeles).

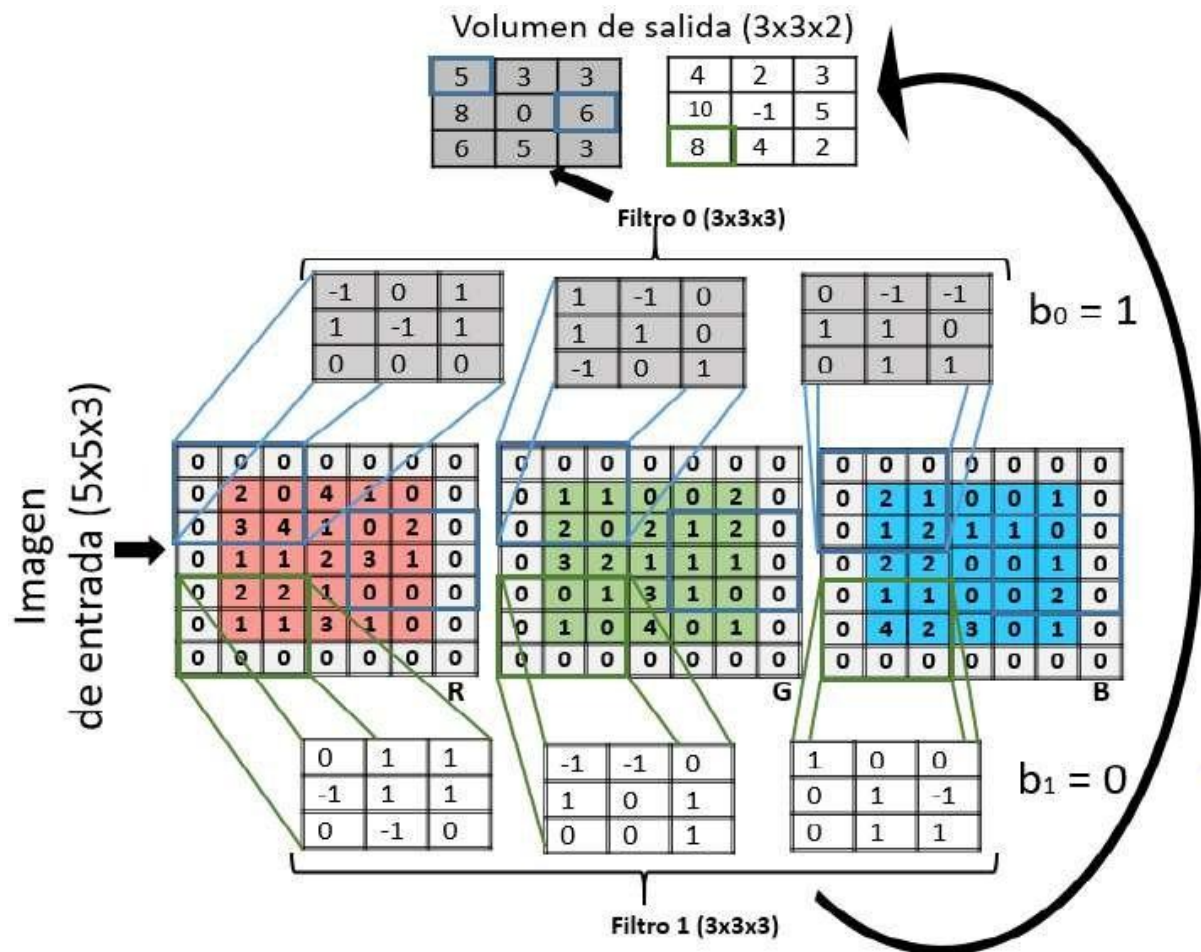
Algoritmo tentativo:

Inicialmente, se tendrá un modelo de capas, siendo la primera de éstas una capa de entrada que recibirá (a definir) los datos transformados desde un arreglo de píxeles, o imagen, a un conjunto de valores de nodos en un grafo. Posteriormente, y como señala el modelo de red neuronal, se presentarán capas de filtrado y reducción de dimensiones de información mediante métodos de aplicación de filtros y superposición de valores (Convolución), reducción de dimensiones (Pooling), activación de determinadas neuronas (Tentativamente mediante ReLu o Sigmoid) y finalmente definición final de clases de imágenes (Softmax) como se presenta a continuación:

- Convolución:

Permite la identificación de la similitud entre una imagen y un filtro, ambos representados de manera matricial en su planteamiento general, el resultado de la convolución será el producto punto entre cada uno de los valores de ambas matrices (imagen y filtro) por cada superposición del filtro (el cual en general es de menor tamaño que la imagen) sobre la

figura (así mismo, al resultado podrá sumarse un valor b , a fines de estabilizar el resultado, como se explica a continuación).



En el caso de los grafos, y basado en el algoritmo de aprendizaje de las Redes Neuronales, se buscará que los pesos de los valores en los filtros sean objeto de aprendizaje, su valor será aleatorio inicialmente y mediante el algoritmo de Back Propagation (explicado posteriormente) se podrán ajustar dichos valores. Finalmente, también el valor b mencionado anteriormente podrá ser usado y aprendido dinámicamente mediante el entrenamiento del modelo.

Así mismo, manteniendo los principios definidos para el caso de las matrices, se tendrá la siguiente ecuación para describir el valor de un nodo en función de sus entradas (nodos que se dirigen a éste) y los pesos de cada una de dichas conexiones:

$$y = \left(\sum_{i=1}^n x_i w_i \right) + b$$

con x_i siendo el valor del nodo i (nótese que solo se tienen en cuenta los valores de los nodos conectados al nodo actual), w_i el peso de la conexión entre el nodo i y el nodo actual y finalmente b , un valor agregado al producto punto de los pesos y los nodos conexos al nodo actual. Éstos dos últimos parámetros serán aprendidos mediante Deep Learning y conformarán la sección principal del estudio.

- Pooling (Máx y Avg)

Representa un método de reducción de las dimensiones de un conjunto de datos, en el caso de una matriz y de manera similar a la convolución, se tomarán regiones de determinado tamaño de una matriz y se simplificarán a un único valor, ya sea escogiendo el mayor de todos (Max Pooling) o bien promediando su valor (Average Pooling).

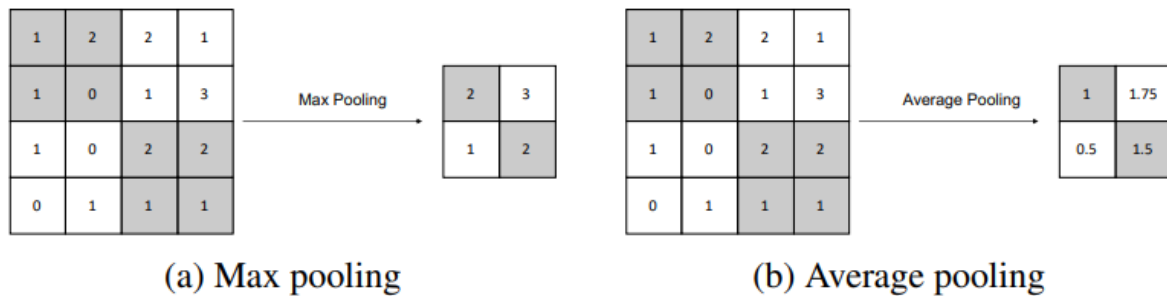


Figure 3.10 Pooling methods in CNNs

Nuevamente, respecto a la implementación en grafos, es posible realizar ésta operación mediante el acceso a los valores de los nodos conexos al nodo actual dentro de una capa de éste tipo, no será relevante además el aprendizaje de los pesos para éste caso ya que no se tendrán en cuenta.

- Funciones de activación

Permiten simular la activación de determinados nodos mediante una función matemática basada en datos anteriores, como objeto del presente proyecto, se realizará tentativamente una conexión directa entre cada nodo de la capa anterior (es decir, conexiones 1 a 1) y se aplicará la correspondiente función de activación:

> ReLU:

$$\text{ReLU}(z) = \max\{0, z\}.$$

> Sigmoid:

$$\sigma(z) = \frac{1}{1 + \exp(-z)}.$$

- Determinación de clases:

Finalmente, y generalmente posicionándose en la última capa, se encuentra una capa de nodos totalmente conexos entre sí que usarán el mismo algoritmo de la capa de convolución:

$$z_i = \left(\sum_{j=1}^m x_j w_j \right) + b$$

para posteriormente calcular la probabilidad final de que cada figura pertenezca a una clase:

$$\hat{y}_i = \text{softmax}(z)_i = \frac{\exp(z_i)}{\sum_j \exp(z_j)}, i = 1, \dots, n$$

donde n representa la cantidad de clases que pueden ser clasificados.

- Backpropagation:

Finalmente, se estudiará éste método, como se mencionó anteriormente, para actualizar los pesos y valores b de las capas de convolución y softmax mientras se entrena el modelo con imágenes y clasificación conocidos.

Cabe resaltar que, el proyecto principalmente será enfocado hacia el estudio de una representación diferente del algoritmo de clasificación de imágenes mediante Deep Learning, principalmente, en contraste a la clásica representación del modelo mediante multiplicación de matrices. Así mismo, la capa de Convolución recibirá un enfoque especial respecto al estudio realizado dada su complejidad y relevancia para la totalidad del modelo, sin embargo, se procurará estudiar su efecto sobre las capas superiores de Pooling, Activación y Softmax.

Referencias:

- Wikipedia.org. (18/08/2022). Convolución. Recuperado de: <https://es.wikipedia.org/wiki/Convoluci%C3%B3n> en 20/10/2022
- Researchgate. (Enero, 2018). Flores A. José M. Deep Learning para la Detección de Peatones y Vehículos. Recuperado de: https://www.researchgate.net/figure/Figura-210-Ejemplo-de-convolucion-con-dos-filtros-de-3x3-en-una-imagen-RGB-de-5x5_fig3_323460108 en 20/10/2022.
- Cambridge University Press. Yao Ma y Jiliang Tang. (2021). Deep Learning on Graphs. Recuperado de: https://web.njit.edu/~ym329/dlg_book/ en 20/10/2022.