Apuntes Semana 11 - 08/05/2025

Deylan Sandoval Sanchez - 2020234274
Escuela de Ingeniería en Computación
Instituto Tecnológico de Costa Rica

I. Arquitecturas convolucionales

A. Composición

* Convolutional Layer: Encargadas de extraer las características de las imagenes

* Pooling Layer: Procesamiento para reducir el tamaño de la imagen

* Dense Layer: Tomar las características extraídas y hacer un clasificador

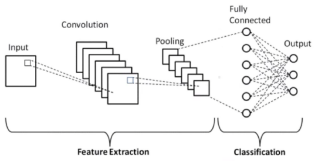

Fig. 1. Composición

$INPUT \rightarrow [[Cconv - relu] * nPool?] * m \rightarrow [fc \rightarrow relu] * k \rightarrow fc$

$3 \geq n \geq 0, m \geq 0, k \geq 0$

Stacks

B. ¿Que arquitectura preferimos?

Prefiera stacks con convoluciones pequeñas

Las neuronas sera computadas con una función lineal directamente con la entrada,en el otro caso el stack contiene no linealidad (RELU) que hacen los features mas expresivos.

Supongamos que todos los volúmenes tienen C canales, la convolución 7x7 tendrá C x (7 x 7 x C) = $49C^3$ parámetros,El stack tendria (3 x (C x (3 x 3 x c))) = $27C^2$

C. Algunas Reglas

* El tamaño de la capa de entrada (contiene la imagen) deberia ser divisible por 2 muchas veces

* Las capas de convoluciones deberia usar campos receptivos pequeños 3x3 o a lo mucho 5x5, utilizando stride = 1

* Padding de ceros de manera que no se altere el espacio

* Configuraciones compunes de maxpooling es F = 2,S = 2 o F = 3, S = 2

Se empieza con una base de datos que va a contener aproximadamente 60.000 muestras donde cada una son imágenes de 28x28 pixeles,cada pixel sera un feature por lo tanto se tiene 784 features, en cada imagen se encuentra un dígito escrito a mano, por lo que cada dígito va a ser diferente aunque sea el mismo numero, el objetivo va a ser tratar de hacer un clasificador de números escritos a mano

D. Arquitecturas

* LeNet

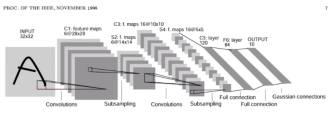

Fig. 2. Arquitectura LeNet

* AlexNet

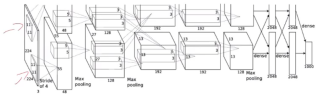

Fig. 3. Arquitectura AlexNet

* ZFnet

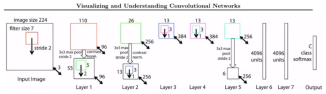

Fig. 4. Arquitectura ZFNET

* GoogleNet

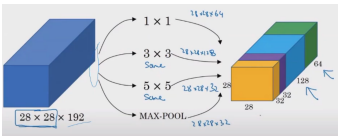

Fig. 5. Arquitectura GoogleNet

* VGG16

Fig. 6. Arquitectura VGG16

* ResNet

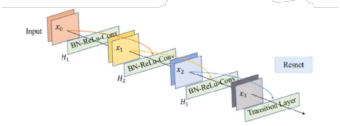

Fig. 7. Arquitectura ResNet

* DenseNet

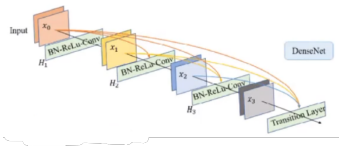

Fig. 8. Arquitectura DenseNet

E. Visualizando que aprenden las redes convolucionales

:

1) Problemas de las redes neuronales: * Dificiles de explicarla

* Features aprendidos no son interpretables

2) Tecnicas para analizas las redes: :

* Visualizar las activaciones:Se muestran las activaciones al momento de la pasada hacia adelante, se pueden ver la neuronas muertas(ReLu)

* Visualizar los filtros: Se visualizan los pesos, se interpretan mejor en la primera capa convolucional y muestran los filtros sin patrones de ruidos

F. Embeddings

* Gradualmente transformamos imágenes a una interpretación(vector), sus clases son separables por un clasificador

* Se pueden transformar a una representación de dimensión menor manteniendo distancias iguales que la representación mayor, t-SNE (paper para la tecnica)

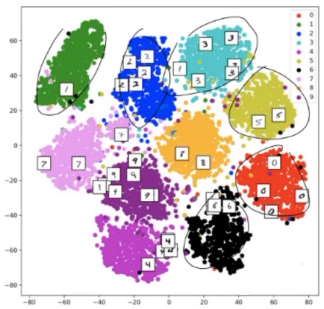

Fig. 9. Embeddings

II. Autoencoder

Se basan en la reconstrucción de una imagen con un vector de características, es supervisando pero con la misma imagen de entrada, por eso se entra en un debate y se clasifica como no-supervisado

Esta compuesto por:

* Encoder: Extrae la características de la imagen

* Espacio latente: Distribuye las características de la imagen

* Decoder: Recontruye la imagen

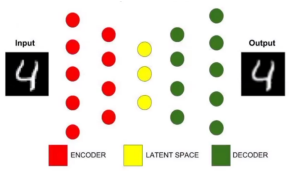

Fig. 10. Autoencoder

¿Que puede hacer un Autoencoder?

* Permite representar la entrada en un vector mas pequeño

* Representación mas poderosa en comparación con PCA

* Permite reconstruir la información con menos perdida